

**PENGEMBANGAN APLIKASI WEB KARYAKU  
SEBAGAI DIREKTORI KARYA MAHASISWA  
UNIVERSITAS NEGERI YOGYAKARTA**

**SKRIPSI**

Diajukan kepada Fakultas Teknik Universitas Negeri Yogyakarta  
untuk Memenuhi Sebagian Persyaratan  
Guna Memperoleh Gelar Sarjana Pendidikan



Oleh:  
**Bintang Muhammad**  
**NIM. 14520241057**

**PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA  
JURUSAN PENDIDIKAN TEKNIK ELEKTRONIKA DAN INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA  
2018**

## **SURAT PERNYATAAN**

Saya yang bertanda tangan di bawah ini:

Nama : Bintang Muhammad  
NIM : 14520241057  
Program Studi : Pendidikan Teknik Informatika  
Fakultas : Teknik  
Judul TAS : Pengembangan Aplikasi Web Karyaku Sebagai Direktori  
Karya Mahasiswa Universitas Negeri Yogyakarta

menyatakan bahwa skripsi ini benar-benar karya saya sendiri. Sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain kecuali sebagai acuan kutipan dengan mengikuti tata penulisan karya ilmiah yang telah lazim.

Yogyakarta, 13 Juli 2018

Yang menyatakan,



Bintang Muhammad

## LEMBAR PERSETUJUAN

Tugas Akhir Skripsi dengan Judul

**PENGEMBANGAN APLIKASI WEB KARYAKU  
SEBAGAI DIREKTORI KARYA MAHASISWA  
UNIVERSITAS NEGERI YOGYAKARTA**

Disusun Oleh:

Bintang Muhammad

NIM 14520241057

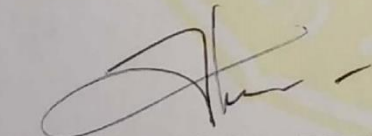
telah memenuhi syarat dan disetujui oleh Dosen Pembimbing untuk dilaksanakan  
Ujian Tugas Akhir Skripsi bagi yang bersangkutan.

Yogyakarta,

2018

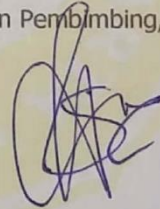
Mengetahui,  
Ketua Program Studi  
Pendidikan Teknik Informatika

Disetujui,  
Dosen Pembimbing,



Handaru Jati, S.T., M.M., M.T., Ph.D.

NIP. 19740511 199903 1 002



Nurkhamid, S.Si., M.Kom., Ph.D.

NIP. 19680707 199702 1 001

**HALAMAN PENGESAHAN**  
Tugas Akhir Skripsi  
**PENGEMBANGAN APLIKASI WEB KARYAKU**  
**SEBAGAI DIREKTORI KARYA MAHASISWA**  
**UNIVERSITAS NEGERI YOGYAKARTA**




Disusun Oleh:

Bintang Muhammad

NIM 14520241057

Telah dipertahankan di depan Tim Penguji Tugas Akhir Skripsi Program Studi  
Pendidikan Teknik Informatika Fakultas Teknik Universitas Negeri Yogyakarta  
pada tanggal 19 Juli 2018

**TIM PENGUJI**

Nama/Jabatan	Tanda Tangan	Tanggal
Nurkhamid, S.Si., M.Kom., Ph.D. Ketua Penguji/Pembimbing		30-07-2018
Dr. Rahmatul Irfan M.T. Sekretaris		30-07-2018
Handaru Jati ST., M.M., M.T., Ph.D. Penguji		30-07-2018

Yogyakarta, 2018

Fakultas Teknik Universitas Negeri Yogyakarta

Dekan,



Dr. Widarto, M.Pd.

NIP. 19631230 198812 1 001



## HALAMAN MOTTO

*“If you're nothing without this suit, then you shouldn't have it.”*

*-Tony Stark to Peter Parker, Spider-man Homecoming (2017)-*

## HALAMAN PERSEMBAHAN

Segala puji bagi Allah SWT, Rabb semesta alam yang senantiasa memberikan karunia sehingga penulis mampu menyelesaikan penulisan skripsi ini. Karya ini saya persembahkan kepada:

1. Orang tua dan keluarga saya yang telah membimbing dan membantu saya dalam hidup ini.
2. Teman-teman PTI F 2014 yang senantiasa semangat dalam menjalani kegiatan kuliah bersama.
3. Teman-teman Infinite UNY yang telah menyediakan kelapangan untuk berbagi ilmu dan tempat bernaung.
4. KKN B97 dan Warga Gude 1 yang telah memberikan dukungan dan doa kepada kelancaran studi saya.
5. *Squad* PLT dan PPL SMK N 2 Magelang yang telah menginspirasi saya selama ini.
6. Mas Deni, Mas Heru, Mas Fauzi, Mas Rais, dan Mas David yang telah menginspirasi dalam dunia TI.
7. Nur Ayuningtias Safitri atas inspirasi, motivasi, pembelajaran dan kesempatan yang berharga selama ini.
8. Kamu yang membaca skripsi ini untuk berjuang lulus, semoga skripsi ini bermanfaat untuk skripsimu.

**PENGEMBANGAN APLIKASI WEB KARYAKU  
SEBAGAI DIREKTORI KARYA MAHASISWA  
UNIVERSITAS NEGERI YOGYAKARTA**

Oleh:

Bintang Muhammad

14520241057

**ABSTRAK**

Karya adalah hasil belajar yang merupakan bagian dari portofolio selama menempuh hidup. Karya juga merupakan hasil buatan, perbuatan maupun ciptaan dari manusia selama proses hidupnya. Mahasiswa selama menempuh pendidikan di tingkat universitas telah banyak menghasilkan karya-karya secara tulis maupun non tulis. Penelitian ini bertujuan untuk (1) Mengembangkan aplikasi berbasis web sebagai media informasi yang digunakan untuk mempublikasikan hasil karya mahasiswa secara daring (2) Menjamin tingkat kualitas perangkat lunak, aplikasi berbasis web dengan melakukan pengujian yang memenuhi standar ISO 25010.

Jenis penelitian "Pengembangan Aplikasi Web Karyaku sebagai Direktori Karya Mahasiswa Universitas Negeri Yogyakarta" adalah Research and Development (R&D). Metode pengembangan *Rational Unified Process* (RUP). RUP memiliki 4 tahapan yaitu: *Inception*, *Elaboration*, *Construction*, dan *Transition*.

Hasil dari penelitian ini adalah (1) Aplikasi web Karyaku telah memenuhi kebutuhan untuk menjadi media informasi publikasi hasil karya mahasiswa secara daring ditunjukkan dengan kesesuaian antara kebutuhan pengguna (*requirement*) dengan fitur yang dikembangkan. (2) Aplikasi web Karyaku sebagai direktori karya mahasiswa telah teruji menggunakan Standar ISO 25010 yang terdiri dari 5 aspek. Hasil pengujian aspek *functional suitability* mendapatkan nilai 1 atau fungsi berjalan benar secara keseluruhan, aspek *usability* dinyatakan sangat layak mendapatkan nilai 85,67%, aspek *performance efficiency* terpenuhi dengan waktu pemuatan 0,91 detik, aspek *reliability* dinyatakan memenuhi syarat dengan rata-rata keberhasilan 99,91%, dan aspek *maintainability* dinyatakan mudah dirawat dengan maintainability index 125,71.

Kata Kunci: publikasi karya, aplikasi website, ISO 25010

## **KATA PENGANTAR**

Puji syukur atas kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul "Pengembangan Aplikasi Web Karyaku Sebagai Direktori Karya Mahasiswa Universitas Negeri Yogyakarta" ini dengan baik.

Tugas Akhir Skripsi ini dapat diselesaikan tidak lepas dari bantuan dan kerja sama dengan pihak lain. Berkenaan dengan hal tersebut, penulis menyampaikan terima kasih kepada:

1. Nurkhamid, S.Si., M.Kom., Ph.D., selaku dosen pembimbing skripsi yang telah memberikan bimbingan dan pengarahan selama pengerjaan skripsi.
2. Handaru Jati, Ph.D., selaku validator instrumen penelitian TAS yang memberikan saran perbaikan secara komprehensif terhadap TAS ini.
3. Ketua Penguji, Penguji Utama, dan Sekretaris yang memberikan koreksi perbaikan secara komprehensif terhadap Tugas Akhir Skripsi ini.
4. Dr. Fatchul Arifin, S.T., M.T., selaku Ketua Jurusan Pendidikan Teknik Elektronika dan Informatika serta dosen dan staff yang telah memberikan bantuan dan fasilitas selama proses penyusunan pra pro proposal sampai dengan selesainya TAS ini.
5. Dr. Widarto, M.Pd., selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta yang memberikan persetujuan pelaksanaan TAS ini.
6. Mahasiswa PTI F 2014 yang telah menemani perkuliahan di kelas selama 3 Tahun.

7. Jumiwati, Anjasromo Adi Nugroho, Taufik Anwar, Permadi Afrian, Arif Sulisty, Linda, Setyorini, dan yang lainnya yang tidak saya sebut satu persatu atas pemberian semangat dan petunjuk pengerjaan.
8. Teman-teman Jurusan Pendidikan Teknik Elektronika dan Informatika atas bantuannya dalam membantu pengerjaan TAS ini mulai dari doa, semangat, hingga menemani pengerjaannya berada di segala tempat.
9. Responden yang telah membantu kelancaran pengambilan data dan saran yang telah dimasukan untuk kemajuan penulisan ini.
10. Infinite, TIM IT Restek UNY, atas tempat bernaungnya selama pengerjaan skripsi ini.
11. Semua pihak yang terlibat atas bantuan dan dukungan selama pengerjaan Tugas Akhir Skripsi ini.

Demikian, guna perbaikan skripsi ini dan karya tulis ilmiah selanjutnya, penulis mengharapkan kritik dan saran yang bersifat membangun dari pembaca. Semoga Tugas Akhir Skripsi ini bermanfaat bagi pembaca.

Yogyakarta, .....

Penulis,

Bintang Muhammad

NIM. 14520241057

## DAFTAR ISI

	Halaman
SURAT PERNYATAAN .....	ii
LEMBAR PERSETUJUAN.....	iii
HALAMAN PENGESAHAN .....	iv
HALAMAN MOTTO .....	v
HALAMAN PERSEMBAHAN .....	vi
ABSTRAK .....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR .....	xii
DAFTAR TABEL.....	xv
DAFTAR LAMPIRAN.....	xvii
BAB I.....	1
A. Latar Belakang Masalah.....	1
B. Identifikasi Masalah .....	4
C. Batasan Masalah.....	4
D. Rumusan Masalah.....	5
E. Tujuan .....	5
F. Spesifikasi Produk .....	5
G. Manfaat Penelitian .....	6
BAB II .....	7
A. Kajian Teori.....	7
1. Pengembangan Aplikasi Web.....	7
2. Analisis Kualitas Perangkat Lunak Aplikasi Web .....	26
B. Penelitian yang Relevan.....	35
C. Kerangka Pikir .....	37
D. Pertanyaan Penelitian.....	38
BAB III .....	40
A. Jenis Penelitian .....	40
B. Prosedur Penelitian .....	40

C. Sumber Data/Subjek Penelitian .....	44
D. Variabel Penelitian .....	44
E. Metode Penelitian dan Alat Pengumpul Data .....	44
F. Instrumen Penelitian .....	45
G. Teknik Analisis Data .....	51
BAB IV .....	55
A. Hasil Penelitian .....	55
1. Pengembangan Aplikasi Web.....	55
a. Tahap Inception.....	55
b. Tahap Elaboration .....	59
c. Tahap Construction .....	74
d. Tahap Transition.....	85
2. Analisis Kualitas Aplikasi Web.....	86
a. Functional Suitability .....	86
b. Usability.....	88
c. Performance Efficiency .....	90
d. Reliability .....	92
e. Maintainability .....	93
B. Pembahasan.....	93
BAB V .....	96
A. Kesimpulan .....	96
B. Keterbatasan Produk.....	97
C. Pengembangan Produk Lebih Lanjut.....	97
D. Saran.....	97
DAFTAR PUSTAKA .....	98
LAMPIRAN .....	102



## DAFTAR GAMBAR

Gambar 1. Lapisan Rekayasa Perangkat Lunak (Pressman, 2014).....	10
Gambar 2. 4 fase dalam Rational Unified Process (Wazlawick, 2014, hal. 4).....	12
Gambar 3. Survei oleh W3Techs tentang penggunaan bahasa pemrograman sisi server (W3Techs, 2018).....	16
Gambar 4. Diagram UML (Fowler, 2004).....	19
Gambar 5. Antarmuka StarUML.....	22
Gambar 6. Bagan konsep MVC secara sederhana (Surguy, 2014) .....	24
Gambar 7. Aspek kualitas perangkat lunak ISO 25010 (Wagner, 2013) .....	28
Gambar 8. 5 aspek pengujian menurut Olsina (Lowe & Pressman, 2009).....	29
Gambar 9. Contoh USE Questionner (Perlman, 2015).....	32
Gambar 10. Kerangka Pikir .....	37
Gambar 11. Alur Karya Mahasiswa didokumentasikan dan dipublikasikan .....	56
Gambar 12. Use Case Diagram Mahasiswa dan Pengunjung .....	62
Gambar 13. Use Case Diagram Administrator.....	65
Gambar 14. Class Diagram Aplikasi .....	69
Gambar 15. Activity Diagram Unggah Karya.....	70
Gambar 16. Sequence Diagram Unggah Karya.....	71
Gambar 17. Desain ERD Basis Data Aplikasi.....	72
Gambar 18. Desain Tampilan Lihat Karya .....	73
Gambar 19. Potongan kode routes .....	75
Gambar 20. Potongan kode model .....	76
Gambar 21. Potongan kode Controller .....	76
Gambar 22. Potongan kode view .....	77
Gambar 23. Implementasi Tampilan Halaman Login .....	78
Gambar 24. Implementasi Halaman Register .....	78
Gambar 25. Implementasi Halaman Jelajah.....	79
Gambar 26. Implementasi Halaman Karya.....	80
Gambar 27. Implementasi Halaman Karya dengan Galeri .....	80
Gambar 28. Implementasi Halaman Sunting Karya .....	81
Gambar 29. Implementasi Halaman Sunting Karya bagian Gallery .....	82
Gambar 30. Implementasi Halaman Karya bagian Video.....	82
Gambar 31. Implementasi Halaman Profil.....	83
Gambar 32. Implementasi Halaman Dashboard Admin.....	84
Gambar 33. Implementasi Halaman Laporan Admin .....	84
Gambar 34. Konfigurasi Heroku .....	85
Gambar 35. Proses pengunggahan dan ekstraksi source code pada hosting .....	86
Gambar 36. Hasil Pengujian Usability.....	89
Gambar 37. Hasil Pengujian Performance Efficiency Halaman Depan .....	91
Gambar 38. Hasil Pengujian Reliability Menggunakan WAPT .....	92

Gambar 39. Hasil Pengujian Maintainability Menggunakan PHPMetrics .....	93
Gambar 40. Hasil Pengujian Maintainability dengan PHPMetrics .....	93
Gambar 40. Activity Diagram Login .....	129
Gambar 41. Activity Diagram Register .....	129
Gambar 42. Activity Diagram Jelajah Karya .....	130
Gambar 43. Activity Diagram Pencarian Karya .....	130
Gambar 44. Activity Diagram Lihat Karya .....	131
Gambar 45. Activity Diagram Laporkan Karya .....	131
Gambar 46. Activity Diagram Sunting Informasi Karya .....	132
Gambar 47. Activity Diagram Sunting Informasi Profil .....	132
Gambar 48. Activity Diagram Unggah Gambar .....	133
Gambar 49. Activity Diagram Tambah Video .....	133
Gambar 50. Activity Diagram Moderasi Karya .....	134
Gambar 51. Activity Diagram Moderasi Pengguna .....	134
Gambar 52. Activity Diagram Tambah Tags .....	135
Gambar 53. Activity Diagram Tambah Prodi .....	135
Gambar 54. Activity Diagram Tambah Admin .....	136
Gambar 55. Activity Diagram Hapus Laporan .....	136
Gambar 56. Activity Diagram Login Admin .....	137
Gambar 57. Sequence Diagram Login .....	138
Gambar 58. Sequence Diagram Jelajah Karya .....	138
Gambar 59. Sequence Diagram Pencarian Karya .....	139
Gambar 60. Sequence Diagram Lihat Karya .....	139
Gambar 61. Sequence Diagram Laporkan Karya .....	139
Gambar 62. Sequence Diagram Sunting Informasi Karya .....	140
Gambar 63. Sequence Diagram Unggah Gambar .....	140
Gambar 64. Sequence Diagram Tambah Video .....	140
Gambar 65. Sequence Diagram Sunting Informasi Profil .....	141
Gambar 66. Sequence Diagram Ganti Password .....	141
Gambar 67. Sequence Diagram Login Admin .....	142
Gambar 68. Sequence Diagram Moderasi Karya .....	142
Gambar 69. Sequence Diagram Moderasi Pengguna .....	142
Gambar 70. Sequence Diagram Tambah Tag .....	143
Gambar 71. Sequence Diagram Tambah Prodi .....	143
Gambar 72. Sequence Diagram Tambah Admin .....	143
Gambar 73. Sequence Diagram Hapus Admin .....	144
Gambar 74. Desain Tampilan Halaman Register .....	145
Gambar 75. Desain Tampilan Halaman Beranda .....	145
Gambar 76. Desain Tampilan Halaman Unggah Karya .....	146
Gambar 77. Desain Tampilan Halaman Pencarian .....	146
Gambar 78. Desain Tampilan Halaman Jelajah Karya .....	147
Gambar 79. Desain Tampilan Halaman Karya .....	147

Gambar 80. Desain Tampilan Halaman Sunting Karya Bagian Informasi Dasar.	148
Gambar 81. Desain Tampilan Halaman Sunting Karya Bagian Galeri .....	148
Gambar 82. Desain Tampilan Halaman Sunting Karya Bagian Video.....	149
Gambar 83. Desain Tampilan Halaman Pelaporan Karya.....	149
Gambar 84. Desain Tampilan Halaman Profil.....	150
Gambar 85. Desain Tampilan Halaman Profil.....	150
Gambar 86. Desain Tampilan Halaman Login Administrator .....	151
Gambar 87. Desain Tampilan Halaman Dashboard Administrator .....	151
Gambar 88. Desain Tampilan Halaman Kelola Laporan.....	152
Gambar 89. Desain Tampilan Halaman Kelola Tags .....	152
Gambar 90. Desain Tampilan Halaman Kelola Program Studi.....	153
Gambar 91. Desain Tampilan Halaman Kelola Adminsitrator .....	153
Gambar 92. Implementasi Halaman Depan .....	154
Gambar 93. Implementasi Halaman Pencarian.....	154
Gambar 94. Implementasi Halaman Sunting Profil .....	155
Gambar 95. Implementasi Halaman Dashboard Admin.....	155
Gambar 96. Implementasi Halaman Kelola Tag .....	156
Gambar 97. Implementasi Halaman Kelola Program Studi .....	156
Gambar 98. Implementasi Halaman Kelola Administrator .....	157
Gambar 99. Hasil Pengujian Performance Efficiency Halaman Login.....	165
Gambar 100. Hasil Pengujian Performance Efficiency Halaman Regsiter .....	165
Gambar 101. Hasil Pengujian Performance Efficiency Halaman Beranda.....	165
Gambar 102. Hasil Pengujian Performance Efficiency Halaman Unggah Karya..	166
Gambar 103. Hasil Pengujian Performance Efficiency Halaman Pencarian .....	166
Gambar 104. Hasil Pengujian Performance Efficiency Halaman Jelajah .....	166
Gambar 105. Hasil Pengujian Performance Efficiency Halaman Karya .....	167
Gambar 106. Hasil Pengujian Performance Efficiency Halaman Sunting Karya..	167
Gambar 107. Hasil Pengujian Performance Efficiency Pelaporan Karya .....	167
Gambar 108. Hasil Pengujian Performance Efficiency Halaman Profil .....	168
Gambar 109. Hasil Pengujian Performance Efficiency Halaman Sunting Profil...	168
Gambar 110. Hasil Pengujian Performance Efficiency Halaman Login Admin ....	168
Gambar 111. Hasil Pengujian Performance Efficiency Halaman Dashboard .....	169
Gambar 112. Hasil Pengujian Performance Efficiency Halaman Kelola Laporan	169
Gambar 113. Hasil Pengujian Performance Efficiency Halaman Kelola Tag .....	169
Gambar 114. Hasil Pengujian Performance Efficiency Halaman Kelola Prodi.....	170
Gambar 115. Hasil Pengujian Performance Efficiency Kelola Administrator.....	170

## DAFTAR TABEL

Tabel 1. 5 Aspek pengujian menurut Olsina .....	29
Tabel 2. Perbandingan Standar Olsina dan ISO 25010 .....	30
Tabel 3. Tabel Instrumen pengujian aspek functional suitability.....	46
Tabel 4. Instrumen aspek usability .....	49
Tabel 5. Analisis Kuantitatif Jawaban skala Likert menurut Sugiyono (2006). ....	52
Tabel 6. Tabel kriteria Interpretasi Nilai .....	53
Tabel 7. Maintainability Index .....	54
Tabel 8. Penjadwalan pengerjaan aplikasi web .....	58
Tabel 9. Spesifikasi laptop yang digunakan untuk mengembangkan aplikasi. ....	58
Tabel 10. Definisi Aktor pada Use Case Diagram .....	60
Tabel 11. Deskripsi Use Case Diagram Mahasiswa dan Pengunjung.....	63
Tabel 12. Skenario Use Case Unggah Karya .....	64
Tabel 13. Deskripsi Use Case Diagram Admin.....	65
Tabel 14. Skenario Use Case Moderasi Karya .....	67
Tabel 15. Tabel Deskripsi Desain Basis Data.....	72
Tabel 16. Daftar Desain Tampilan Aplikasi .....	74
Tabel 17. Penguji Functional Suitability .....	87
Tabel 18. Hasil Pengujian Functional Suitability .....	87
Tabel 19. Tabel Hasil Pengujian Performance Efficiency .....	91
Tabel 20. Hasil Pengujian Reliability menggunakan WAPT .....	95
Tabel 21. Skenario Use Case Login.....	112
Tabel 22. Skenario Use Case Register.....	112
Tabel 23. Skenario Use Case Jelajah Karya .....	113
Tabel 24. Skenario Use Case Pencarian Karya .....	113
Tabel 25. Skenario Use Case Lihat Karya .....	114
Tabel 26. Skenario Use Case Bagikan Karya .....	115
Tabel 27. Skenario Use Case Laporkan Karya .....	115
Tabel 28. Skenario Use Case Sunting Informasi Karya.....	116
Tabel 29. Skenario Use Case Unggah Gambar .....	117
Tabel 30. Skenario Use Case Hapus Gambar .....	117
Tabel 31. Skenario Use Case Tambah Video.....	118
Tabel 32. Skenario Use Case Hapus Video .....	119
Tabel 33. Deskripsi Use Case Sunting Informasi Profil.....	119
Tabel 34. Skenario Use Case Ganti Gambar Profil .....	120
Tabel 35. Skenario Use Case Ganti Password .....	121
Tabel 36. Skenario Use Case Login Admin.....	121
Tabel 37. Skenario Use Case Tambah Tag .....	122
Tabel 38. Skenario Use Case Hapus Tag .....	123

Tabel 39. Skenario Use Case Tambah Admin.....	123
Tabel 40. Skenario Use Case Hapus Admin .....	124
Tabel 41. Skenario Use Case Sunting Password Admin.....	124
Tabel 42. Skenario Use Case Tambah Prodi .....	125
Tabel 43. Skenario Use Case Sunting Prodi .....	125
Tabel 44. Skenario Use Case Hapus Prodi .....	126
Tabel 45. Skenario Use Case Moderasi Pengguna .....	127
Tabel 46. Skenario Use Case Hapus Laporan.....	127

## DAFTAR LAMPIRAN

Lampiran 1. Surat Keputusan Pembimbing Skripsi .....	104
Lampiran 2. Lembar Persetujuan Proposal Skripsi .....	106
Lampiran 3. Surat Pernyataan Validasi Instrumen .....	107
Lampiran 4. Hasil Angket Survey Kepemilikan Karya Mahasiswa.....	108
Lampiran 5. Skenario Use Case .....	112
Lampiran 6. Activity Diagram .....	129
Lampiran 7. Sequence Diagram .....	138
Lampiran 8. Desain (wireframe) Antarmuka Aplikasi. ....	145
Lampiran 9. Implementasi Antarmuka Aplikasi .....	154
Lampiran 10. Hasil Pengujian Functional Suitability .....	158
Lampiran 11. Hasil Pengujian Usability.....	161
Lampiran 12. Hasil Pengujian Performance Efficiency .....	165
Lampiran 13. Foto Kegiatan .....	171

## **BAB I**

### **PENDAHULUAN**

#### **A. Latar Belakang Masalah**

Karya adalah hasil belajar yang merupakan bagian dari portofolio selama menempuh hidup. Karya juga merupakan hasil buatan, perbuatan maupun ciptaan dari manusia selama proses hidupnya. Mahasiswa selama menempuh pendidikan di tingkat universitas telah banyak menghasilkan karya-karya secara tulis maupun non tulis. Banyaknya mata kuliah yang mengharuskan mahasiswa untuk membuat produk sebagai syarat tugas akhir atau untuk menempuh ujian semester membuat setiap mahasiswa menghasilkan karya-karya. Jurusan Pendidikan Teknik Elektronika dan Informatika memiliki beberapa mata kuliah yang ditawarkan untuk masing-masing prodi beberapa mata kuliah yang mewajibkan pembuatan produk sebagai bentuk tugas yang akan dinilai untuk pertimbangan nilai akhir. Mahasiswa biasanya mengumpulkan karya-karya mereka hanya berbentuk laporan dan hanya tampil saat presentasi dan tidak ditampilkan secara luas di luar aktivitas perkuliahan. Beberapa mahasiswa memiliki produk yang bagus untuk di pameran.

Karya juga merupakan bagian dari portofolio yang berguna untuk mencari pekerjaan. Karya bisa menjadi pertimbangan bagi sebuah perusahaan untuk melamar bekerja karena akan digunakan sebagai bukti kemampuan yang nyata dari mahasiswa tersebut. Karya sebagai dokumentasi apa yang telah dikerjakan oleh mahasiswa selama berada di perkuliahan. Mahasiswa diharuskan mendata karya-karya yang telah dibuatnya serta mendeskripsikannya dengan secara detail. Karya sebaiknya juga dipublikasi secara luas sehingga masyarakat luas mengetahui karya yang telah dibuat dan meminimalkan plagiasi terjadi. Pada saat ini, para



mahasiswa masih belum mendokumentasikan karya-karya yang mereka miliki. Mahasiswa sering membiarkan karya yang telah dibuat berakhir di ujian akhir saja tidak didokumentasikan secara baik. Mahasiswa juga belum mempublikasikan karya mereka secara daring dan terpusat. Mahasiswa tidak mempunyai media publikasi yang cocok untuk karya-karya mereka.

Menurut Vina Georgiana (2011), program studi pada universitas memerlukan sarana teknologi elektronik yang berguna untuk mendokumentasikan serta mempublikasikan karya mahasiswa. Teknologi elektronik tersebut membantu pihak universitas dan mahasiswa mempublikasikan karya mereka secara daring. Universitas Negeri Yogyakarta khususnya Jurusan Pendidikan Teknik Elektronika dan Informatika belum memiliki website yang digunakan untuk menampilkan galeri karya-karya mahasiswa dalam bentuk gambar. Maka dari itu Jurusan Pendidikan Teknik Elektronika dan Informatika membutuhkan sebuah gallery sebagai media informasi yang terhubung secara daring yang dapat digunakan untuk sarana publikasi dan dokumentasi hasil karya mahasiswa. Selain itu, karya-karya mahasiswa bisa dijadikan dokumentasi untuk kepentingan jurusan maupun tingkat universitas jika dibutuhkan.

Menurut survey dilakukan oleh Asosiasi Penyelenggara Jasa Internet Indonesia (APJII) pada tahun 2016, jumlah penetrasi internet 2017 sebanyak 88,1 juta jiwa dari 252, 4 juta jiwa masyarakat Indonesia atau sekitar 34,9% dari seluruh masyarakat Indonesia. Ini membuktikan bahwa pengguna internet sudah meningkat secara pesat di Indonesia. Dari survey tersebut, jumlah terbesar pengguna internet adalah masyarakat yang berentang umur dari 10 hingga 35 tahun sebanyak 42,8% dari total seluruh penetrasi internet. Dalam jumlah

pengguna internet di Indonesia, memberi peluang mempublikasikan informasi karya-karya yang dibuat oleh mahasiswa Universitas Negeri Yogyakarta kepada masyarakat luas. Dengan adanya peluang tersebut, diperlukan adanya sarana publikasi karya-karya mahasiswa yang terpusat di skala Universitas. Aplikasi website karyaku dapat menjadi salah satu solusi sebagai direktori karya mahasiswa yang dapat diakses secara daring. Aplikasi ini dikembangkan dengan menggunakan website yang memungkinkan diakses secara daring, langsung, dan dari mana pun.

Pengujian adalah sebuah hal penting dalam pengembangan perangkat lunak berbasis website. Pengujian perangkat lunak berbasis website mencegah terjadinya beberapa risiko kesalahan secara teknis maupun non teknis yang dilakukan oleh pengguna. Perangkat lunak berbasis website yang kurang teruji kualitasnya dapat timbul permasalahan waktu dalam memuat sebuah halaman. Risiko tidak dilakukannya pengujian yang lain seperti kesalahan informasi yang didapatkan oleh pengguna karena kesalahan yang dilakukan oleh perangkat lunak. Dari hal-hal yang sudah disampaikan mengenai pengujian perangkat lunak berbasis website bisa dijadikan acuan untuk melakukan serangkaian pengujian dalam melakukan pengembangan aplikasi website agar memiliki kualitas yang sesuai dengan standar. Berbagai macam standar pengujian yang banyak digunakan antara lain: McCall, Boehm, FURPS, Start, BBN, ISO 9126, dan ISO 25010. Dalam penelitian ini, ISO 25010 digunakan sebagai standar yang akan digunakan dalam pengujian supaya mengetahui dan menguji kualitas karyaku sebagai direktori karya mahasiswa Universitas Negeri Yogyakarta.

## **B. Identifikasi Masalah**

Berdasarkan latar belakang yang telah dipaparkan oleh peneliti maka dapat diidentifikasi permasalahan sebagai berikut:

1. Jurusan Pendidikan Teknik Elektronika dan Informatika memiliki beberapa mata kuliah yang mengharuskan mahasiswa membuat karya sebagai syarat tugas akhir untuk menempuh ujian atau tugas akhir.
2. Mahasiswa kurang baik dalam mendokumentasikan karya-karya mereka yang telah dibuat.
3. Mahasiswa masih kurang dalam melakukan publikasi karya-karya mereka secara luas sehingga hanya terbatas di mata kuliah tersebut.
4. Keterbatasan media informasi yang digunakan untuk mempublikasikan hasil karya mahasiswa secara daring.
5. Jumlah pengguna internet di Indonesia sebanyak 88,1 juta jiwa atau sekitar 34,9% dari seluruh masyarakat Indonesia yang secara efektif untuk mempublikasikan karya mahasiswa secara daring.
6. Kurangnya kualitas perangkat lunak khususnya aplikasi berbasis web yang dapat menimbulkan kesalahan teknis maupun non teknis yang dilakukan oleh pengguna maupun kesalahan aplikasi.

## **C. Batasan Masalah**

Dalam penelitian ini, peneliti akan membatasi permasalahan yang akan diselesaikan. Permasalahan yang dibatasi oleh penulis adalah sebagai berikut:

1. Keterbatasan media informasi yang digunakan untuk mempublikasikan hasil karya mahasiswa secara daring.

2. Kurangnya kualitas perangkat lunak khususnya aplikasi berbasis web yang dapat menimbulkan kesalahan teknis maupun non teknis yang dilakukan oleh pengguna maupun kesalahan aplikasi.

#### **D. Rumusan Masalah**

Berdasarkan batasan masalah yang telah ditentukan oleh penulis, maka dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana cara mengembangkan aplikasi web untuk mengatasi keterbatasan media informasi yang digunakan untuk mempublikasikan hasil karya mahasiswa?
2. Bagaimana menjamin kualitas perangkat lunak khususnya aplikasis berbasis web supaya tidak menimbulkan kesalahan?

#### **E. Tujuan**

Tujuan dari penelitian ini sesuai untuk menjawab rumusan masalah yang telah dipaparkan oleh peneliti sebelumnya, sebagai berikut:

1. Mengembangkan aplikasi berbasis web sebagai media informasi yang digunakan untuk mempublikasikan hasil karya mahasiswa secara daring.
2. Menjamin tingkat kualitas perangkat lunak, aplikasi berbasis web dengan melakukan pengujian yang memenuhi standar ISO 25010.

#### **F. Spesifikasi Produk**

1. Pengunjung dapat menjelajah karya yang sudah ada.
2. Pengguna dapat mengunggah karya.
3. Pengguna dapat menambah gambar pendukung untuk karya yang telah diunggah.

4. Pengguna dapat menempelkan *video* untuk karya yang telah diunggah.
5. Pengguna dapat melihat karya mahasiswa lain.
6. Pengguna dapat mencari karya mahasiswa lain.
7. Administrator dapat memoderasi karya dan mahasiswa.
8. Administrator dapat memoderasi *tag* karya.

## **G. Manfaat Penelitian**

Manfaat dari penelitian ini adalah sebagai berikut:

### **1. Manfaat Teoritis**

Manfaat teoritis yang didapat dari penelitian ini adalah sebagai berikut:

- a. Penelitian ini dapat digunakan sebagai referensi bagi peneliti lain yang hendak melakukan penelitian yang relevan
- b. Penelitian ini dapat menambah wawasan keilmuan mengenai konsep pengembangan perangkat lunak khususnya aplikasi berbasis web.

### **2. Manfaat Praktis**

Manfaat praktis dari penelitian pengembangan aplikasi web karya yaitu membantu mahasiswa dan civitas akademika untuk memudahkan mencari dan mempublikasikan karya yang ada di Universitas Negeri Yogyakarta.

## **BAB II**

### **KAJIAN TEORI**

#### **A. Kajian Teori**

##### **1. Pengembangan Aplikasi Web**

##### **a. Perangkat Lunak Aplikasi Web**

Menurut Rosa A.S & Shalahuddin (2011, hal. 4) Perangkat Lunak adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan. Definisi perangkat lunak menurut Pressman (2014) perangkat lunak adalah kumpulan intruksi-intruksi atau program komputer yang ketika program dieksekusi menyediakan fitur, fungsi dan kinerja yang diinginkan.

Menurut Rosa A. S. dan M. Shalahudin (2011, hal. 2) Perangkat lunak memiliki beberapa karakteristik sebagai berikut:

- 1) Perangkat Lunak dibangun dengan rekayasa bukan diproduksi secara manufaktur atau pabrikan.
- 2) Perangkat Lunak tidak pernah usang karena kesalahan atau kecacatan dalam perangkat lunak tersebut dapat diperbaiki.
- 3) Barang produksi pabrikan biasanya komponen barunya akan terus diproduksi, sedangkan perangkat lunak biasanya terus diperbaiki seiring bertambahnya kebitihan.

Menurut Pressman (2014) terdapat kategori-kategori perangkat lunak komputer, yaitu:

1) Perangkat Lunak Sistem

Perangkat Lunak Sistem adalah sebuah koleksi dari tulisan program untuk memenuhi kebutuhan program lain. Contoh dari Perangkat Lunak Sistem adalah: *compiler, editor*, dan pengorganisasi berkas.

2) Perangkat Lunak Aplikasi

Perangkat lunak aplikasi merupakan perangkat lunak yang berdiri sendiri dan bertujuan untuk menyelesaikan kebutuhan bisnis khusus. Aplikasi pada lingkungan kegiatan bisnis atau data teknis bertujuan untuk memfasilitasi operasi bisnis atau memanajemeni pengambilan keputusan.

3) Perangkat Lunak Rekayasa/Saintifik

Perangkat Lunak Rekasayasa adalah perangkat lunak yang memiliki algoritma yang terkait dengan keilmuan dari cakupan ilmu astronomi hingga vulkanologi. Perangkat lunak ini digunakan untuk melakukan perhitungan-perhitungan khusus yang rumit.

4) Perangkat Lunak Tertanam

Perangkat lunak tertanam adalah perangkat lunak yang ada diantara produk dan sistem dan digunakan untuk mengimplementasikan dan mengontrol fitur dan fungsi kepada para pengguna dan sistem itu sendiri. Perangkat lunak tertanam bisa melakukan kegiatan terbatas atau menyediakan fungsi signifikan dan kemampuan kontrol (seperti: fungsi digital pada *automobile*).

5) Perangkat Lunak Lini Produk

Perangkat lunak lini produk merupakan perangkat lunak yang menyediakan sebuah kemampuan khusus untuk digunakan oleh beberapa pelanggan yang



berbeda. Perangkat lunak lini produk ini fokus terhadap tempat belanja terbatas seperti kontrol produk inventori atau alamat pelanggan yang banyak.

#### 6) Aplikasi Web atau Perangkat Bergerak

Aplikasi web dan perangkat bergerak merupakan perangkat lunak yang berorientasi secara jaringan (*network-centric software*) menjangkau beragam aplikasi dan dapat mencakup aplikasi yang diakses melalui peramban (*browser-based*) dan aplikasi yang terdapat pada perangkat bergerak itu sendiri.

#### 7) Perangkat Lunak Kecerdasan Buatan

Perangkat lunak kecerdasan buatan merupakan perangkat lunak yang menggunakan algoritma non-numerik untuk menyelesaikan permasalahan yang rumit yang tidak bisa diterima untuk dikomputasi atau analisis biasa. Perangkat lunak ini diaplikasikan pada lingkup robotika, sistem pakar, pengenalan pola gambar dan suara, jaringan saraf tiruan, dan permainan.

Aplikasi web merupakan sebuah sistem perangkat lunak berbasis teknologi dan standar *World Wide Web Consortium* (W3C) yang menyediakan sumber daya seperti konten dan pelayanan melalui sebuah tampilan antarmuka yang diakses melalui *web browser* (Kappel, Prohl, & Reich, 2003). Lowe dan Pressman (2009, hal. 4) menyatakan aplikasi web merupakan sebuah perangkat lunak karena berisi koleksi atau kumpulan intruksi dan data yang menyediakan informasi dan fungsionalitas untuk pengguna. Aplikasi web merupakan bagian dari perangkat lunak sehingga disiplin yang digunakan untuk mengembangkan aplikasi web sama dengan perangkat lunak.

## b. Rekayasa Perangkat Lunak Aplikasi Web

Rekayasa perangkat lunak merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai, dapat dipercaya, dan bekerja efisien dengan mesin (Rosa A.S. & Shalahuddin, 2011). Menurut Pressman (2014), Rekayasa perangkat lunak merupakan sebuah aplikasi dari yang sistematis, disiplin, dapat dihitung dalam pendekatan pengembangan, operasi dan perawatan perangkat lunak.

Rekayasa perangkat lunak memiliki 4 lapisan. Lapisan-lapisan tersebut bertingkat seperti gambar 1.



Gambar 1. Lapisan Rekayasa Perangkat Lunak (Pressman, 2014)

Pondasi dari rekayasa perangkat lunak adalah lapisan proses (*process layer*). Proses rekayasa perangkat lunak merekatkan lapisan teknologi menjadi satu dan memungkinkan pengembangan yang rasional dan terukur secara waktu dalam pengembangan perangkat lunak komputer. Metode rekayasa perangkat lunak (*methods*) menyediakan cara teknis dalam pembuatan perangkat lunak. Metode menunjukkan proses pembuatan perangkat lunak dari beberapa tugas yang terdiri dari komunikasi, analisis kebutuhan, desain model, konstruksi program, pengujian, dan dukungan. *Tools* atau peralatan dalam rekayasa perangkat lunak menyediakan dukungan otomatis maupun semi-otomatis dalam proses dan metode. Ketika peralatan terintegrasi maka informasi yang dibuat oleh sebuah alat bisa digunakan

oleh alat lain, sebuah sistem yang mendukung pengembangan perangkat lunak, atau bisa disebut *computer-aided software engineering*, yang sudah terpasang.

Aplikasi web merupakan bagian dari perangkat lunak sehingga untuk pengembangannya termasuk proses rekayasa perangkat lunak yang membutuhkan proses, metode, dan peralatan. Menurut Roger Pressman (2009, hal. 16), prinsip-prinsip yang ada pada rekayasa perangkat lunak bisa diaplikasikan ke dalam pengembangan aplikasi web. Jadi pengembangan aplikasi web dapat menggunakan model pengembangan perangkat lunak yang lain.

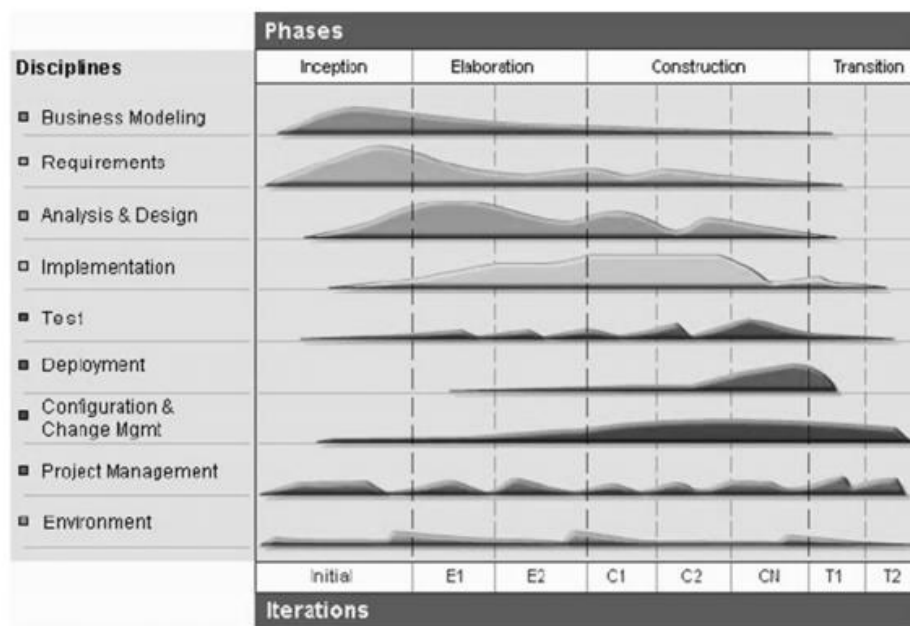
### **c. Model Pengembangan Perangkat Lunak Aplikasi Web**

Model pengembangan perangkat lunak yang akan digunakan untuk mengembangkan aplikasi web ini adalah *Rational Unified Process* (RUP). RUP merupakan sebuah pendekatan pengembangan yang dilakukan secara berulang atau iteratif, fokus pada arsitektur dan lebih diarahkan kepada penanganan kasus (Rosa A.S. & Shalahuddin, 2011, hal. 105). RUP merupakan turunan dari *Unified Process* yang dicirikan iteratif dan incremental. Iteratif bisa dilakukan pada setiap tahap untuk menghasilkan perbaikan lebih baik (*incremental*), dimana setiap iterasi akan memperbaiki proses berikutnya. Model pengembangan RUP menurut Rosa A. S. dan M. Shalahuddin (2011, hal. 111) memiliki beberapa keuntungan dibandingkan dengan model pengembangan waterfall adalah sebagai berikut:

- 1) RUP mampu mengakomodasi perubahan kebutuhan perangkat lunak yang dapat ditambah atau dikurangi sesuai dengan keinginan.
- 2) Integrasi bukanlah sebuah proses besar dan cepat pada diakhir proyek karena dengan model pengembangan yang iteratif.
- 3) Risiko yang biasanya ditemukan selama pada proses integrasi awal.

- 4) Kecacatan dan kesalahan dapat diperbaiki secara pada beberapa iteratif sehingga menghasilkan arsitektur yang lebih baik dan aplikasi menjadi berkualitas tinggi.
- 5) Pengembangan perangkat lunak dapat diperbaiki seiring proses pengembangan perangkat lunak.

RUP memiliki 4 tahap yang terdapat pada Gambar 2.



Gambar 2. 4 fase dalam Rational Unified Process (Wazlawick, 2014, hal. 4)

#### 1) *Inception*

Tahap inception merupakan tahap permulaan. Tahap ini fokus pada pemodelan proses bisnis (*business modeling*), pendefinisian kebutuhan untuk sistem yang akan dibuat (*requirement*), penentuan jadwal proyek (*project management*), dan penentuan lingkungan (*environment*) pengembangan yang akan digunakan (Kruchten, 2004). Tahap *inception* bertujuan untuk memahami

lingkup proyek, kasus bisnis yang perlu diselesaikan, dan risiko-risiko yang akan terjadi.

*Business modeling* merupakan sebuah aktivitas untuk mengumpulkan informasi mengenai aktivitas-aktivitas yang biasa dilakukan untuk dijadikan otomatisasi atau strukurisasi ulang (Wazlawick, 2014, hal. 7). *Business modeling* bisa dilakukan menggunakan observasi dengan *stakeholders*, melakukan wawancara terhadap target pengguna, atau melakukan studi mengenai dokumen-dokumen. Menurut Kruchten (2004, hal. 157), *business modeling* bertujuan untuk pemetaan struktur dan kedinamisan dari organisasi yang akan ada dalam sistem, mengetahui permasalahan dan potensi yang akan di selesaikan dengan produk, memastikan pengguna dan pengembang memiliki pemahaman yang sama, dan memperoleh kebutuhan sistem untuk mendukung penyelesaian masalah.

*Requirement analysis* merupakan aktivitas menganalisis kondisi dan kemampuan sistem yang akan dikembangkan berdasarkan model bisnis yang telah dilakukan (Kruchten, 2004). Aktivitas ini fokus terhadap pengolahan model bisnis yang telah didapatkan lalu diubah menjadi kebutuhan sistem yang akan dibuat. *Requirement* dapat dibedakan menjadi 2 yaitu *functional requirements* dan *nonfunctional requirements* (Kruchten, 2004). *Functional requirements* yang mengarah ke fungsional fitur-fitur yang dibutuhkan oleh sistem, sedangkan *nonfunctional requirements* yang mengarah ke hasil dan kualitas pada fitur-fitur yang akan dikembangkan. *Requirement analysis* menghasilkan pemahaman yang lebih baik kepada pengembang terhadap *requirement*, definisi batasan-batasan sistem, serta dasar-dasar perkiraan waktu dan biaya untuk mengembangkan sistem.

*Project management* adalah aktivitas untuk menyeimbangkan pekerjaan untuk mendapatkan tujuan, manajemen risiko, dan mengatasi batasan waktu untuk menjadikan produk yang sesuai dengan kebutuhan pengguna (Kruchten, 2004, hal. 131). *Project management* menganalisis waktu yang dibutuhkan untuk setiap tahap pengembangan dalam RUP. *Project management* mengatur cara untuk menyelesaikan produk secara tepat waktu dan hemat biaya. Penjadwalan adalah kunci dari *project management* untuk membatasi waktu untuk tiap tahapan dalam RUP supaya tidak terjadi penggunaan waktu yang berlebih.

*Environment* merupakan sebuah aktivitas penentuan untuk mendukung pengembangan produk atau aplikasi dengan peralatan yang mendukung. Hal yang diperhatikan dalam penentuan alat pendukung yaitu: pemilihan peralatan atau tools yang tepat untuk dukungan aplikasi, pemasangan peralatan, dan konfigurasi dari *tools itu* sendiri. Menurut Lowe (Lowe & Pressman, 2009) dalam pengembangan aplikasi website, *environment* yang digunakan dalam pengembangan harus menggunakan teknologi yang sama dengan *environment* yang digunakan dalam produksi namun dengan konfigurasi sedikit berbeda disesuaikan dengan pengembangan. *Environment* yang digunakan dalam proses pengembangan dan produksi adalah sebagai berikut:

a) PHP

Menurut Frank M. Kromann (2016), PHP merupakan bahasa pemrograman scripting yang biasa digunakan untuk membuat konten web yang dinamis. Hal ini juga didukung oleh Hariyanto (2008) yang mengatakan bahwa PHP merupakan bahasa pemrograman sisi server yang digunakan untuk membangun website dinamis. Terdapat 80 persen dari web server terpasang PHP dalam berbagai

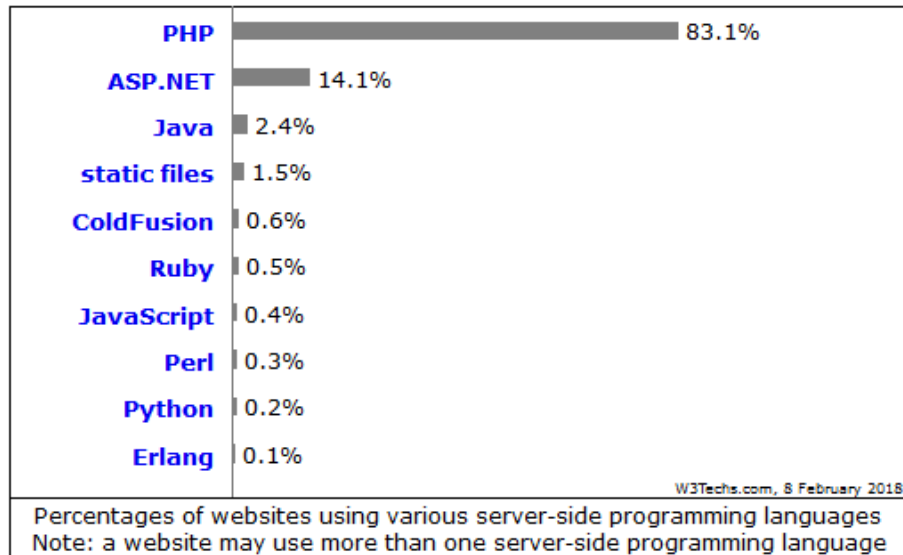
bentuk. Bahasa pemrograman PHP pertama kali dibuat oleh Rasmus Ledrof pada tahun 1994. Pada awalnya, PHP merupakan akronim dari Personal Home Page. Tujuan awal Rasmus membuat PHP adalah untuk mendukung sebuah halaman web sederhana. Pada tahun akhir 1990, Zeev Suraski dan Andi Gutman yang berasal dari Zend Technologies mengembangkan PHP dengan menulis ulang kode dan menghasilkan PHP versi 3. PHP versi 3 memberikan akronim baru pada PHP yaitu *PHP: Hypertext Preprocessor*.

Posisi PHP terdapat pada urutan teratas dalam 10 bahasa pemrograman di dunia. Dilansir dari TIOBE yang merupakan perusahaan yang bergerak dibidang kualitas perangkat lunak, PHP berada di urutan ke 6. Dalam dekade terakhir, setelah PHP 5 rilis di juli 2004, PHP telah dikenal sebagai solusi populer untuk membangun aplikasi web (Ajzele, 2016 ).

Bahasa pemrograman PHP merupakan bahasa *scripting* yang bebas karena tidak perlu mendeklarasikan variabel sebelum menggunakannya dan variabel dapat berubah tipe datanya selama konten juga berganti (Kromann, 2016). Beberapa platform populer seperti WordPress, Drupal, Magentom dan Prestashop dibangun menggunakan PHP dan mendukung pertumbuhan popularitas bahasa pemrograman PHP. Beberapa platform tersebut telah mengembangkan batas dari PHP dengan mengimplementasi pola Pemrograman Berbasis Objek (PBO) yang terdapat pada bahasa pemrograman lain seperti Java, C#, dan lain-lain. Implementasi PBO inilah yang mendasari pengembangan PHP versi selanjutnya untuk mendukung PBO dalam PHP. Pada tahun 2010, pengembangan PHP versi 6 dihentikan. Pada tahun 2015, Dmitry Stogov, CPE dari Zend Technologies mengumumkan pengembangan PHP versi 7 dan dirilis pada Desember 2015.



Penggunaan PHP versi 7 pada tahun 2018 telah mencapai 9,7 persen dari 83,1 persen seluruh website yang ada di dunia yang terdapat pada Gambar 3 (W3Techs, 2018).



Gambar 3. Survei oleh W3Techs tentang penggunaan bahasa pemrograman sisi server (W3Techs, 2018)

#### b) MySQL Database

Menurut Irmawati dan Indrihapsari (2014), *database* merupakan kompetetisasi yang bertujuan untuk memelihara dan membuat informasi tersedia pada saat diperlukan. Menurut Antonio Lopes (2016, p. 133), Database merupakan alat untuk manajemen data. Fungsi dasar database adalah memasukkan (*inserting*), mencari (*searching*), memperbarui (*updating*), dan menghapus (*deleting*) data. untuk kepentingan administrasi data, program komputer yang biasanya dimanfaatkan untuk pengembangan antara lain MySQL (*Structured Query Language*) dan Microsoft Access (Asiatun, 2011).

MySQL merupakan *Relational Database Systems* (RDBMS) karena menggunakan SQL sebagai bahasa untuk berkomunikasi dengan sistem. Menurut

Paul DuBois (2016), MySQL merupakan database management yang populer dengan berbagai alasan seperti cepat, mudah dalam pemasangan dan penggunaan, dan mudah diadministrasi. Selain itu, MySQL dapat berjalan di berbagai macam sistem operasi seperti UNIX dan Windows. Program yang berbasis MySQL bisa ditulis diberbagai macam bahasa.

Menurut Dyer (2015, hal. 1) MySQL adalah server. Kata server disini dalam konteks ke perangkat lunak, bukan sebuah komputer utama yang aktif. Server yang dimaksud adalah melakukan pemeliharaan, pengontrolan, dan perlindungan terhadap data, meletakkan data tersebut pada komputer yang menjalankan server dalam berbagai bentuk. MySQL sebagai server mendengarkan beberapa permintaan dari perangkat lunak lain yang sedang berjalan yang pada konteks ini dianggap sebagai client. Client yang dianggap disini adalah dalam konsep perangkat lunak, bukan komputer. Sebuah client dan server mungkin bisa di dalam satu komputer yang sama.

Hasil akhir dari tahap *inception* adalah *business modeling, requirements, project management* dalam bentuk jadwal untuk tiap tahap dalam RUP, dan *environment* berupa *tools* yang akan digunakan selama pengembangan dan produksi.

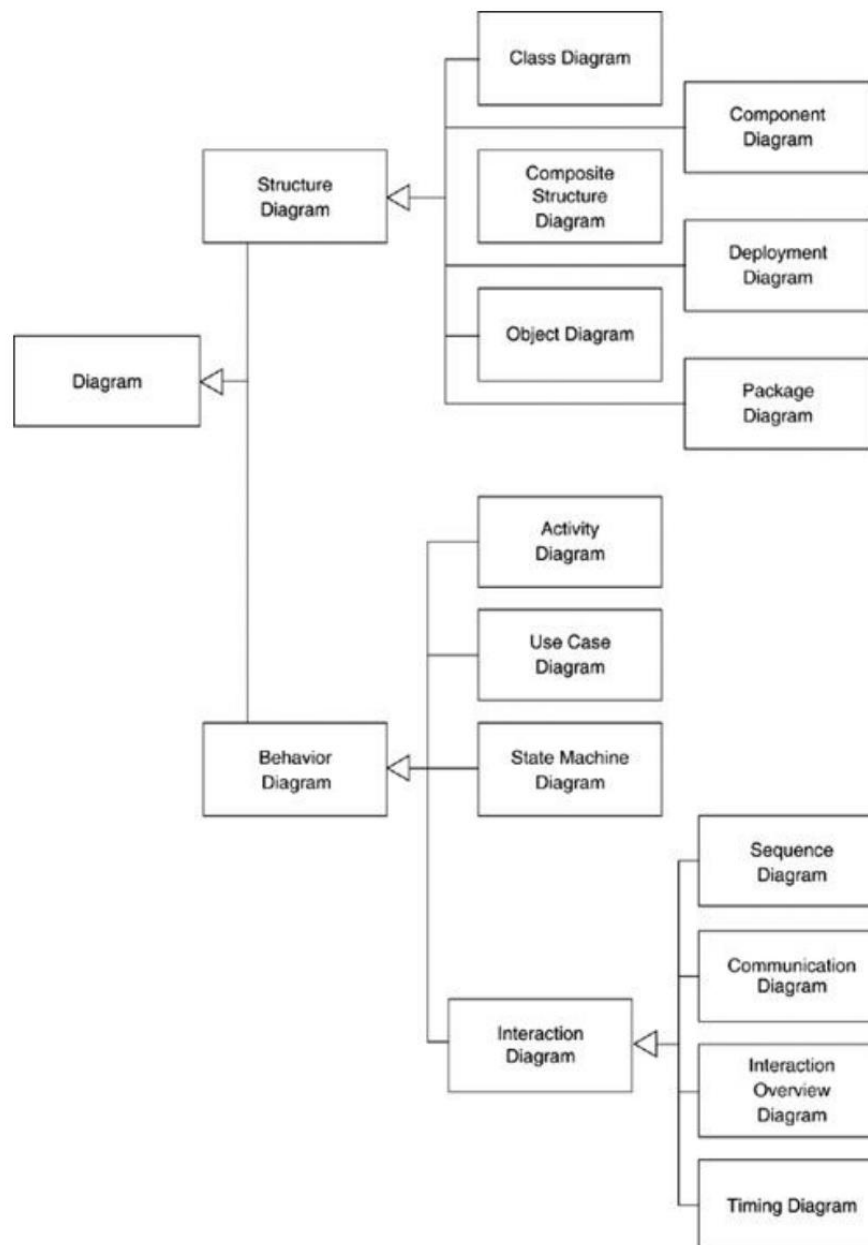
## 2) *Elaboration*

Tahap *elaboration* fokus pada perancangan sistem yang akan dibuat (Rosa A.S. & Shalahuddin, 2011, hal. 110). Tahap *elabortation* lebih berfokus kepada disiplin *analysis and design*. *Analysis and design* merupakan proses menerjemahkan kebutuhan sistem yang telah didapatkan menjadi spesifikasi yang mendeskripsikan cara untuk diimplementasikan dalam sistem (Kruchten, 2004).

Pada aktivitas ini berfokus pada pemodelan sistem yang akan dikembangkan. Menurut Rosa A. S. dan M. Shalahuddin (2011, hal. 116), Pemodelan adalah gambaran realita yang sederhana dan dituangkan dalam bentuk pemetaan dengan aturan tertentu. Pemodelan dapat menggunakan sama dengan realitas. Pemodelan memudahkan manusia memahami secara visual agar mengerti bagaimana suatu proyek dapat dipahami dan dikerjakan. Pemodelan digunakan untuk meminimalisasi kegagalan dan risiko yang akan terjadi. Salah perangkat pemodelan perangkat lunak adalah *Unified Modeling Language*.

*Unified Modeling Language* atau yang disebut UML merupakan bahasa pemodelan visual yang memiliki tujuan umum yang digunakan untuk menetapkan, memvisualisasikan, membangun, dan mendokumentasi artifak-artifak dari sebuah sistem perangkat lunak. (Rumbaugh, Jacobson, & Booch, 2004, hal. 3). Menurut Martin Fowler (2004, hal. 1), UML merupakan keluarga nota grafis yang didukung oleh meta-model tunggal yang digunakan untuk membantu pendeskripsian dan desain sistem perangkat lunak yang khususnya dibangun menggunakan konsep pemrograman berorientasi objek (OO). UML hanya bisa digunakan dalam pemodelan sehingga tidak terbatas pada metodologi tertentu (Rosa A.S. & Shalahuddin, 2011, hal. 120). UML dikembangkan oleh Rumbaugh dan Booch pada tahun 1991. Pada tahun 1996, pengembang UML mengajukan proposal mengenai standarisasi pemodelan berorientasi objek dan diresmikan pada September 1997 (Rosa A.S. & Shalahuddin, 2011, hal. 120). UML versi 2.0 dikembangkan lagi pada tahun 2000 hingga 2003 untuk mengatasi permasalahan yang belum terkuak pada pengalaman penggunaan dan memperluas kapabilitas domain aplikasi (Rumbaugh, Jacobson, & Booch, 2004, hal. 3).

UML memiliki 13 diagram yang dibagi 2 antara *structure diagram* dan *behaviour diagram* dijelaskan pada gambar 5. Pada penelitian ini, hanya 4 diagram UML yang akan digunakan antara lain: *use case*, *class diagram*, *sequence diagram*, dan *activity diagram*.



Gambar 4. Diagram UML (Fowler, 2004)

a) *Use Case*

Use case adalah sebuah teknik untuk merekam persyaratan fungsional sebuah sistem (Fowler, 2004, hal. 143). Menurut Rosa A. S. dan M. Shalahuddin (2011, hal. 130), *Use Case* merupakan pemodelan kelakuan sistem informasi yang akan dibuat. *Use Case* mendeskripsikan interaksi antara pengguna sistem dengan sistem itu sendiri, dengan memberi narasi yang jelas mengenai sistem tersebut digunakan. *Use case* digunakan untuk mengetahui fungsi yang ada di dalam sebuah sistem dan aktor yang terlibat dalam menggunakan fungsi-fungsi itu.

b) *Class Diagram*

Menurut Rosa A. S. dan M. Shalahuddin (2011, hal. 122), *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat pada sistem yang akan dibangun. Kelas bisa memiliki atribut dan operasi (*method*). Atribut adalah variabel-variabel yang dimiliki oleh kelas, sedangkan operasi adalah fungsi-fungsi yang dimiliki oleh suatu kelas. Sedangkan menurut Martin Fowler (2004, hal. 53) *Class diagram* mendeskripsikan jenis-jenis objek pada sistem dan berbagai macam hubungan statis yang terdapat antara kelas-kelas di sistem yang akan dibangun. *Class diagram* menunjukkan batasan hubungan antar objek yang terdapat pada sistem.

c) *Sequence Diagram*

Menurut Martin Fowler (2004), *sequence diagram* menjabarkan kelakuan sebuah skenario tunggal. Sedangkan menurut Wazlawick (2014), *sequence diagram* merupakan salah satu bagian dari diagram UML yang digunakan untuk mewakili urutan kejadian sistem dan kembali kepada sebuah *use case*. *Sequence diagram* menggambarkan kelakuan sebuah objek dan mendeskripsikan waktu

hidupnya dan pesan yang dikirimkan dan diterima antar objeknya (Rosa A.S. & Shalahuddin, 2011, hal. 137). *Sequence diagram* berguna untuk merangkai langkah-langkah yang akan dilakukan dan respon yang diterima dari sebuah objek untuk menghasilkan informasi tertentu. Elemen yang digunakan pada *sequence diagram* adalah aktor, garis hidup, objek, waktu aktif, dan garis arah.

d) *Activity Diagram*

Menurut Martin Fowler (2004, hal. 81), *activity diagram* merupakan sebuah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja, Diagram ini seperti diagram alir tetapi perbedaan antara *activity diagram* adalah diagram ini mendukung kelakuan yang paralel. *Activity diagram* menggambarkan aktivitas sistem bukan perlakuan yang dilakukan oleh aktor (Rosa A.S. & Shalahuddin, 2011, hal. 81). Diagram ini dapat menggambarkan rancangan proses bisnis dan urutan atau pengelompokan tampilan dari antarmuka jika setiap aktivitas memiliki antarmuka.

Pemodelan sistem menggunakan pemodelan UML dapat menggunakan tools bernama StarUML. StarUML adalah peralatan yang *open source* yang dapat digunakan untuk mengembangkan desain UML. StarUML adalah peralatan yang populer dan sudah diunduh lebih dari 5 juta kali. Perangkat lunak ini dapat menyediakan peralatan yang dapat diinstall di Max OS, Windows, dan Linux sehingga banyak diminati oleh para pengembang perangkat lunak yang menggunakan notasi UML. StarUML telah digunakan oleh perusahaan besar seperti Apple, IBM, Boeing, Amazon, Oracel, Samsung, dan lain-lain. Berikut ini adalah tampilan dari StarUML.



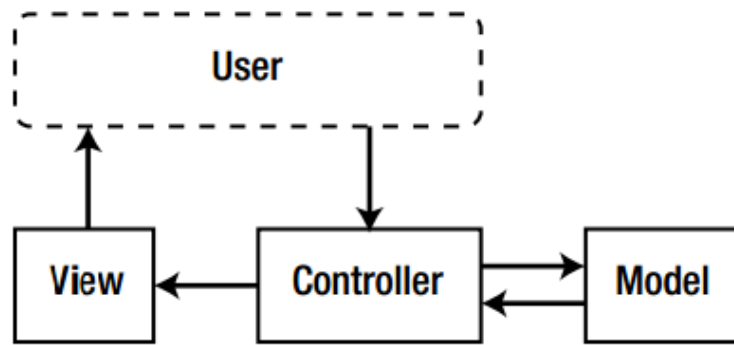
untuk menulis kode tidak hanya logika bisnis yang unik untuk aplikasi yang dikembangkan, tetapi juga setiap komponen yang umum berada semua situs termasuk autentikasi pengguna, validasi masukan, akses basis data, *templating*, dan sebagainya (Stauffer, 2016, hal. 5). Pada zaman sekarang ini, programmer memiliki banyak *framework* aplikasi yang bisa membantu proses pengembangan sebuah perangkat lunak.

Menurut Stauffer (2016, hal. 7), *Framework* adalah kumpulan dari komponen-komponen pihak ketiga yang bersama-sama melekat dengan *custom framework* seperti *configuration file*, *service providers*, *prescribed directory structures*, dan *applications bootstraps*. Keuntungan menggunakan framework adalah mengurangi jumlah kode baru yang pengembang buat pada proyek baru. Keuntungan tersebutlah yang membuat proses pengembangan menjadi lebih efisien secara waktu dan hasil.

Laravel merupakan salah satu dari PHP *Web Application Framework*. Menurut Maksim Surguy (2014, hal. 1) PHP *Web Application Framework* adalah sebuah set atau kumpulan dari kelas, *library*, atau komponen yang ditulis dengan menggunakan bahasa pemrograman PHP yang bertujuan untuk menyelesaikan masalah umum dalam pengembangan aplikasi web dan menggunakan konsep penggunaan kembali kode (*reuse code*).

Laravel merupakan *framework* PHP yang menggunakan konsep *Model*, *View*, dan *Controller*. Konsep MVC merupakan konsep pemrograman yang membuat aplikasi menjadi *three-way factoring* dimana objek dan kelas mengendalikan operasi yang ada di perangkat lunak (Krasner & Stephen, 1988). Konsep MVC secara sederhana ditampilkan pada Gambar 4.





Gambar 6. Bagan konsep MVC secara sederhana (Surguy, 2014)

a) *Model*

Model adalah tempat semua logika bisnis pada aplikasi disimpan (Pitt, 2012). Logika bisnis bisa apa saja yang spesifik seperti cara aplikasi menyimpan data, atau menggunakan layanan pihak ketiga, dalam rangka untuk memenuhi kebutuhan bisnis. Laravel menggunakan eloquent ORM untuk mengendalikan data yang ada. Eloquent ORM menyediakan sebuah implementasi ActiveRecord yang mudah dan cantik dalam mengakses *database* (Laravel, 2018). Dalam Laravel, setiap tabel yang ada dalam *database* menduduki satu *model* yang digunakan untuk berinteraksi dengan tabel tersebut. Model mengizinkan untuk melakukan *query* data dalam tabel anda.

b) *View*

*View* dalam konsep MVC adalah tempat seluruh elemen antarmuka pengguna pada aplikasi disimpan (Pitt, 2012, hal. 1). Menurut Krasner (1988), Views berurusan dengan segala hal yang berhubungan dengan grafis. Kegunaan *view* adalah menampilkan data yang diminta dari *model* dan diolah pada *controller*. Komponen view dapat berupa HTML, Javascript, dan CSS (Pitt, 2012).

Laravel mengendalikan view dengan format bahasa yang disebut Blade. Blade merupakan template *engine* yang digunakan oleh Laravel (Laravel, 2018). Keunggulan Blade adalah tidak melarang penggunaan kode PHP murni dalam *view* karena semua *view* yang menggunakan Blade akan dikompilasi menjadi kode PHP murni dan disimpan sementara hingga terjadi perubahan kode.

c) *Controller*

*Controller* merupakan komponen yang menghubungkan antara *model* dan *view* (Pitt, 2012). Di dalam *controller* terdapat kelas yang digunakan untuk mengelompokkan logika-logika program untuk penanganan *request* (Laravel, 2018). *Request* yang masuk ke *routes* pada Laravel dapat dialihkan ke *controller* untuk diolah menjadi satu tempat. Tugas utama *controller* adalah menangkap HTTP *Request* dan meneruskannya ke aplikasi atau *view* (Stauffer, 2016).

Tahap *construction* juga berisi *testing* untuk menguji kelancaran fungsionalitas atau fitur yang telah dikembangkan dalam sistem. *Testing* yang dilakukan dalam tahap ini yaitu: *unit testing*, *integration testing*, dan *system testing* (Kruchten, 2004). *Unit testing* adalah pengujian hal terkecil yang dapat diuji, jika dalam pengembangan sistem berorientasi objek hal terkecil adalah *methods* dalam *class*. *Integration testing* merupakan pengujian integrasi antar komponen aplikasi web dari program integrasi dengan tampilan yang ada. *System testing* merupakan pengujian aplikasi secara keseluruhan sebagai sebuah sistem yang utuh.

Tahap *construction* menghasilkan produk program yang dikembangkan. Produk yang dikembangkan merupakan aplikasi web yang telah melewati *unit*

*testing, integration testing, dan system testing.* Aplikasi web yang telah dikembangkan sudah siap untuk memasuki tahap *transition*.

#### 4) *Transition*

Tahap *transition* merupakan tahap terakhir dalam model pengembangan RUP. Tahap ini berfokus pada pemasangan atau instalasi sistem agar dapat digunakan oleh pengguna (Rosa A.S. & Shalahuddin, 2011, hal. 111). Selain itu, tahap ini berfokus pada pengujian akhir dan penyampaian sistem kepada pengguna termasuk proses migrasi data (Wazlawick, 2014, hal. 5). Tahapan ini berfokus pada kegiatan *deployment* dan *testing*.

*Deployment* merupakan kegiatan pemasangan sistem yang dikembangkan pada lingkungan operasional. Fokus *deployment* adalah mengantarkan aplikasi web ke pengguna akhir (Lowe & Pressman, 2009). Pemasangan aplikasi web dilakukan pada *production server* yang bisa digunakan target pengguna secara luas. *Deployment* dikatakan berhasil jika pemasangan dilakukan berhasil dan dapat dilakukan *testing*.

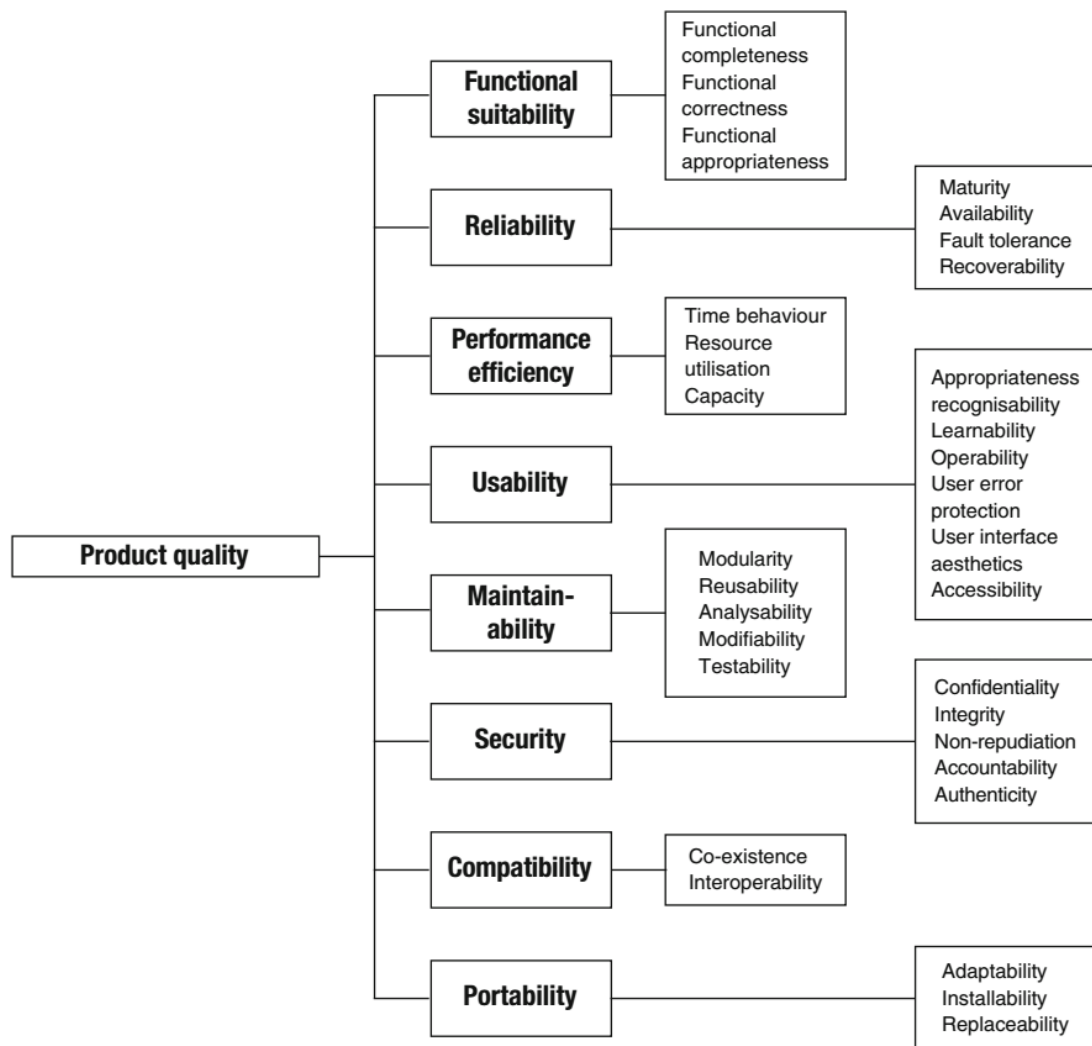
*Testing* pada tahap *transition* berupa *alpha* dan *beta testing* (Kruchten, 2004). *Alpha testing* adalah tahapan pengujian dalam lingkungan pengembangan tertutup yang dilakukan oleh target pengguna yang representatif (Pressman, 2014). *Beta testing* adalah tahapan pengujian sistem kepada target pengguna di lingkungan pengembangan secara terbuka (Sommerville, 2011).

## **2. Analisis Kualitas Perangkat Lunak Aplikasi Web**

Pokok dari manajemen kualitas ditetapkan oleh manufaktur industri dalam mengarahkan untuk meningkatkan kualitas dari produk yang sedang dibuat dan didukung oleh Crosby yang menyatakan kualitas perangkat lunak berdasarkan

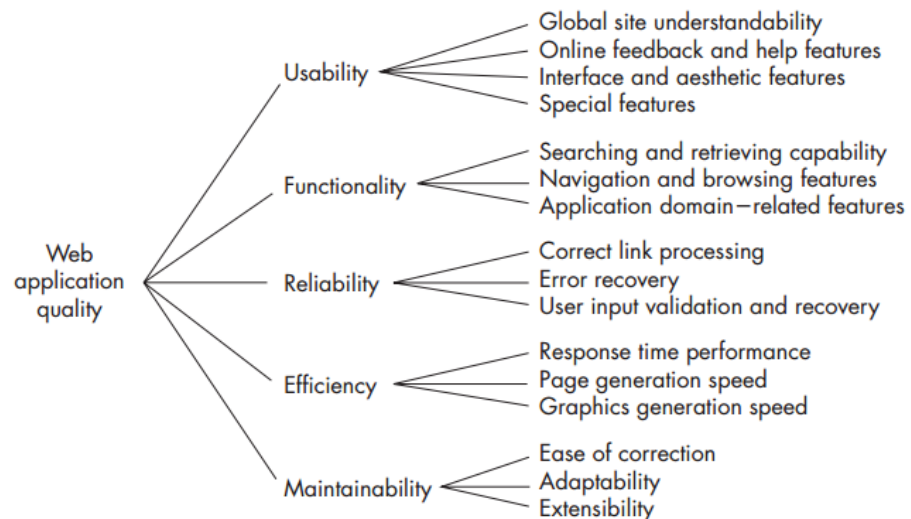
kesesuaian antara hasil dan spesifikasi produk yang detail (Sommerville, 2011). Menurut Rosa A. S dan M. Shalahuddin (2011) menyatakan bahwa perangkat lunak perlu dijaga kualitasnya karena kualitas perangkat lunak bergantung pada kepuasan pengguna. Menurut Pressman (2014), terdapat 3 konsep yang paling penting dalam kualitas perangkat lunak yaitu proses perangkat lunak yang efektif yang mampu mendukung upaya-upaya dalam membangun sebuah perangkat lunak berkualitas tinggi, Produk yang berguna dalam mengantarkan konten, fungsi serta fitur, dan perangkat lunak mampu memberikan nilai tambah pada masing-masing pengembang dan pengguna perangkat lunak itu sendiri.

Dalam pengembangan perangkat lunak terdapat beberapa standar kualitas seperti McCall, Garvin, ISO 9126 dan ISO 25010. ISO 25010 merupakan pembaruan dari Standar ISO 9126 dalam bidang perangkat lunak (Wagner, 2013). Standar ISO 25010 dikembangkan berdasarkan McCall dan Boehm dan membagi pembagian kualitas perangkat lunak menjadi ukuran lebih kecil dengan tujuan terdapat bagian-bagian yang dapat diatur secara mandiri. Dalam ISO 25010 terdapat 8 aspek pengujian yaitu *functional suitability*, *reliability*, *performance efficiency*, *usability*, *security*, *compatibility*, dan *portability*. 8 aspek pengujian tersebut dijabarkan lebih rinci di gambar 7.



Gambar 7. Aspek kualitas perangkat lunak ISO 25010 (Wagner, 2013)

Menurut Olsina dan rekannya (Lowe & Pressman, 2009) standar kualitas aplikasi web ada pada 5 aspek yaitu *functionality*, *usability*, *reliability*, *maintainability* dan *efficiency*. Penjabaran terdapat pada gambar 8 dan tabel 1.



Gambar 8. 5 aspek pengujian menurut Olsina (Lowe & Pressman, 2009)

Tabel 1. 5 Aspek pengujian menurut Olsina

No.	Aspek	Keterangan
1	Usability	Website mudah dipahami secara global
		Terdapat timbal balik dan fitur bantuan
		Antarmuka dan fitur yang berestetika
		Fitur khusus
2	Functionality	Kemampuan dalam melakukan pencarian dan penerimaan
		Fitur navigasi dan penjelajahan
		Fitur yang berhubungan dengan aplikasi
3	Reliability	Pembenahan tautan
		Validasi masukan dan pemulihan
		Pemulihan kesalahan
4	Efficiency	Performa waktu respon
		Kecepatan pengolahan halaman
		Kecepatan pengolahan gambar
5	Maintainability	Kemudahan dalam melakukan koreksi kesalahan
		Kemampuan web dalam adaptasi
		Kemampuan web untuk dikembangkan

Berdasarkan kualitas perangkat lunak menurut ISO 25010 dan standar kualitas web menurut Olsina dkk yang dikutip dari David Lowe (2009). maka peneliti akan mengambil 5 aspek dari standar ISO 25010 yaitu *functional suitability*,

*usability, reliability, performance efficiency, dan maintainability.* Peneliti membandingkan ISO 25010 dengan standar kualitas web yang telah dikemukakan oleh Olsina pada tabel 2.

Tabel 2. Perbandingan Standar Olsina dan ISO 25010

<b>Olsina</b>	<b>ISO 25010</b>
<i>Functionality</i>	<i>Functionality suitability</i>
<i>Usability</i>	<i>usability</i>
<i>Reliability</i>	<i>reliability</i>
<i>efficiency</i>	<i>performance efficiency</i>
<i>maintainability</i>	<i>maintainability</i>

Penjelasan 5 aspek pengujian untuk aplikasi web direktori karya mahasiswa sebagai berikut:

a. *Functional suitability*

Aspek pengujian fungsionalitas untuk mengetahui produk atau sistem telah memenuhi kebutuhan pengguna ketika digunakan. Karakteristik ini dibagi menjadi 3 aspek yaitu:

- 1) *Functional completeness* mengukur sebagaimana fungsi yang telah disediakan mencakup semua tugas dan tujuan pengguna.
- 2) *Functional correctness* mengukur fungsi yang disediakan telah mencakup fungsi yang benar atau tidak.
- 3) *Functional appropriateness* mengukur sejauh mana fungsi yang telah disediakan mampu memfasilitasi penyelesaian tugas dan tujuan tertentu.

*Functional Suitability* diuji oleh ahli pengembangan perangkat lunak ataupun dari pihak *quality control*. *Functional suitability* membutuhkan *test case* sebagai pengujian yang berisi perangkat pengujian untuk menemukan titik kegagalan dalam pengembangan perangkat lunak (Bala & Chhillar, 2016). Skala yang cocok

digunakan untuk *functional suitability* adalah skala Guttman. Skala Guttman merupakan skala yang berdimensi tunggal (Kusumah & Dwigatama, 2009, hal. 80). Skala guttman digunakan untuk jawaban yang sudah jelas dan konsisten seperti ya-tidak, atau yakin-tidak yakin (Sugiyono, 2006, hal. 139). Skala guttman cocok digunakan untuk menguji jumlah fungsional pada aplikasi yang berjalan kemudian dianalisis dengan cara dibandingkan dengan fungsional yang telah direncanakan.

*b. Usability*

Aspek pengujian *usability* merupakan pengujian terhadap aplikasi perangkat lunak berdasarkan kemudahan penggunaan sistem oleh pengguna (Wagner, 2013). *Usability* diuji bertujuan untuk memastikan masing-masing kategori mendukung antarmuka sistem dan dapat mempelajari serta menggunakan semua navigasi dan semantik (Lowe & Pressman, 2009). Aspek pengujian *usability* mempunyai beberapa aspek yang dijelaskan sebagai berikut:

- 1) *Appropriateness recognizability* mengukur seberapa jauh pengguna mengetahui sistem atau produk sesuai dengan kebutuhan pengguna tersebut.
- 2) *Learnability* mengukur seberapa jauh produk atau sistem dapat digunakan oleh pengguna dengan belajar menggunakan sistem tersebut.
- 3) *Operability* mengukur kemudahan pengoperasian suatu sistem atau produk.
- 4) *User error protection* mengukur produk atau sistem melindungi pengguna dalam membuat kesalahan.
- 5) *User interfaces aesthetics* mengukur kemampuan tampilan antarmuka untuk memuaskan interaksi pengguna dengan sistem atau produk.



6) *Accessibility* mengukur seberapa jauh sistem atau produk dapat digunakan oleh semua kalangan untuk mencapai tujuan tertentu atau konteks yang lebih spesifik.

Pengujian *usability* dapat menggunakan USE Questionnaire (Lund, 2001). Pengujian dengan menggunakan USE Questionnaire yang dikembangkan oleh Lund berbentuk skala likert. Menurut Kusumah Wijaya (2009, hal. 79), Skala Likert adalah merupakan sejumlah perangkat pernyataan positif hingga negatif mengenai objek sikap. USE Questionnaire pernah digunakan oleh Perlman pada gambar 9.

**USE Questionnaire: Usefulness, Satisfaction, and Ease of use**

Based on: Lund, A.M. (2001) *Measuring Usability with the USE Questionnaire*. STC Usability SIG Newsletter, 8:2. [\[Abstract\]](#) | [About quest.cgi](#)

Please rate your agreement with these statements.

- Try to respond to all the items.
- For items that are not applicable, use: **NA**
- Make sure these fields are filled in: **System:**   **Email to:**
- Add a comment about an item by clicking on its icon, or add comment fields for all items by clicking on **Comment All**.
- To mail in your results, click on: **Mail Data**

System:    Email to:

Optionally provide comments and your email address in the box.

USEFULNESS		1	2	3	4	5	6	7		NA
1. It helps me be more effective.	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree	<input type="radio"/>
2. It helps me be more productive.	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree	<input type="radio"/>
3. It is useful.	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree	<input type="radio"/>
4. It gives me more control over the activities in my life.	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree	<input type="radio"/>
5. It makes the things I want to accomplish easier to get done.	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree	<input type="radio"/>
6. It saves me time when I use it.	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree	<input type="radio"/>
7. It meets my needs.	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree	<input type="radio"/>
8. It does everything I would expect it to do.	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree	<input type="radio"/>
EASE OF USE		1	2	3	4	5	6	7		NA
9. It is easy to use.	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree	<input type="radio"/>

Gambar 9. Contoh *USE Questionner* (Perlman, 2015)

USE Questionnaire mencakup 4 aspek *usability* dari *usefulness*, *satisfaction*, *ease of learn*, dan *ease of use*. Kuisisioner ini mengandung 30 pertanyaan yang sudah dibagi-bagi menjadi 4 aspek tadi.

c. *Reliability*

Menurut Suryan (2014), Pengujian aspek *reliability* adalah derajat ukur sistem, produk, atau komponen meskipun terdapat kesalahan perangkat keras maupun lunak. *Reliability* ditujukan untuk mengukur sejauh mana produk atau sistem melakukan fungsi tertentu pada kondisi tertentu dalam jangka waktu yang telah ditetapkan. Aspek *reliability* memiliki beberapa sub-aspek sebagai berikut:

- 1) *Maturity* merupakan derajat yang mengukur sistem, produk, atau komponen bertemu dengan kebutuhan *reliability* dibawah operasi normal.
- 2) *Availability* merupakan derajat untuk mengukur kesediaan produk atau sistem ketika dibutuhkan.
- 3) *Fault Tolerance* merupakan derajat pengukuran untuk sistem atau produk dalam menghadapi kesalahan yang dilakukan oleh perangkat keras atau lunak.
- 4) *Recoverability* merupakan derajat untuk mengukur kemampuan sistem melakukan pemulihan data dan kembali ke keadaan awal sistem dalam menghadapi interupsi atau kegagalan.

Pengujian aspek *reliability* dapat dilakukan dengan menggunakan metode *stress testing*. Menurut Debasis Pradhan (2013), *stress testing* pada aplikasi web dapat menggunakan perangkat lunak yang bernama WAPT atau *Web Application Load, Stress and Performance Testing*. WAPT melakukan simulasi dengan menggunakan pengguna tidak nyata untuk mengakses website secara bersamaan dalam waktu yang telah ditentukan.

d. *Performance Efficiency*

Aspek pengujian *performance efficiency* bertujuan untuk mendeskripsikan seberapa baik produk merespon permintaan pengguna dan pelanggan, serta mengukur efisiensi dalam eksekusi permintaan pelanggan (Wagner, 2013). Performa diuji dengan beberapa kondisi, konfigurasi, dan *loading* dalam memastikan respon sistem terhadap interaksi pengguna (Lowe & Pressman, 2009). Aspek *performance efficiency* dibagi menjadi 3 sub sebagai berikut:

- 1) *Time behavior* merupakan derajat yang mengukur waktu respon dan proses ketika sistem melakukan fungsi dan kebutuhannya.
- 2) *Resource Utilization* merupakan derajat yang mengukur besar dan macam sumber daya yang digunakan oleh produk dan sistem ketika melakukan fungsi dan kebutuhannya.
- 3) *Capacity* merupakan derajat yang mengukur batas maksimum dari sebuah sistem.

Pengujian aspek *performance* dapat menggunakan *tools* yang bernama GTMetrix. GTMetrix merupakan sebuah *tools* gratis yang mengevaluasi kecepatan web (Kaur, Kaur, & Kaur, 2016). GTMetrix menganalisis kecepatan halaman web. GTMetrix merupakan gabungan antara Google Page Speed Insight dan Yahoo YSlow, sehingga dengan sekali pengujian bisa mendapatkan 2 hasil sekaligus. GTMetrix digunakan untuk mengukur waktu muat halaman web, GTMetrix juga dapat memberikan rekomendasi agar source code yang diprogram agar lebih efisien.

#### e. *Maintainability*

Aspek *maintainability* merupakan aspek yang mengukur seberapa efisien sebuah produk atau sistem dapat diubah untuk diperbaiki, diadaptasi, atau

dikembangkan pada lingkungan baru. Aspek *maintainability* dibagi menjadi beberapa sub sebagai berikut:

- 1) *Modularity* mengukur derajat sistem atau program komputer terdiri dari komponen-komponen khas sehingga perubahan ke komponen tertentu memiliki dampak minimal ke komponen lain.
- 2) *Reusability* mengukur derajat aset yang dapat digunakan lebih dari 1 sistem.
- 3) *Analyzability* mengukur derajat keefektifan dan kefeisiensiaan dari diagnosis sistem dalam perubahan dari salah satu bagian produk, penyebab kegagalan produk, dan mengidentifikasi bagian yang diubah.
- 4) *Modifiability* mengukur keefektifan pengubahan produk atau sistem tanpa mengganggu atau mengurangi kualitas produk tersbur
- 5) *Testability* mengukur keefektifan pada kriteria pegujian yang bisa dilakukan pada sebuah sistem, produk, atau komponen dan pengujian dapat dilakukan untuk menentukan kriteria yang telah ditemukan.

Maintainability dapat diukur dengan *software metrics*. *Software metrics* digunakan untuk mengukur *Maintainability Index*. MI merupakan indeks ukur yang menunjukkan sebuah angka tunggal yang digunakan untuk mengukur *maintainability* pada sistem (Najm, 2014). *Maintainability Index* pada aplikasi web dapat menggunakan perangkat lunak bernama PHPMetrics. PHPMetrics menganalisis sumber kode PHP yang telah dibuat.

## **B. Penelitian yang Relevan**

1. Aplikasi *Virtual Gallery* Berbasiskan *Content Management System* pada Jurusan Sistem Sistem Informasi Management System oleh Vina Georgiana di Universitas Bina Nusantara pada tahun 2011. Penelitian ini bertujuan

untuk melakukan pengoleksian data karya mahasiswa pada jurusan Sistem Informasi. Aplikasi website ini diberi nama BeeShow. Fungsi dari aplikasi ini mulai dari mengelola grup, mengelola tampilan, mengelola kategori, Mengisi konten, dan mempublikasi content. Hasil penelitian ini adalah aplikasi BeeShow mampu digunakan untuk mendokumentasikan dan mempublikasikan karya mahasiswa secara luas dengan menggunakan website.

2. Perancangan Website sebagai Galeri Program Studi Teknik Informatika D3 Multimedia di Universitas Dian Nuswantoro oleh Abu Salam, M.Kom dan Erwin Fathullah Hariwijaya. Penelitian ini bertujuan untuk menjadikan website galeri untuk mahasiswa program studi Multimedia di Universitas Dian Nuswantoro. Website ini memiliki 3 tingkatan pengguna dari administrator, pengguna biasa, dan anggota. Kelemahan yang dikemukakan oleh penulisnya adalah kurangnya perancangan yang matang pada tampilan website.
3. Implementasi *Framework* Laravel untuk Galeri Seni Berbasis Web oleh Widi Hastom pada tahun 2015 dipublikasikann di Fakultas Teknologi Industri Unissula. Penelitian ini bertujuan untuk mempublikasikan karya seni mahasiswa dan digunakan sebagai media promosi penjualan barang seni. Penelitian in juga memaparkan keuntungan penggunaan *framework* Laravel yang cepat dan efisien dalam pengembangan sistem.

### C. Kerangka Pikir



Gambar 10. Kerangka Pikir

Kerangka pikir yang ada pada Gambar 10 menjelaskan adanya permasalahan keterbatasan media informasi yang digunakan untuk mempublikasikan hasil karya mahasiswa dan rendahnya kualitas perangkat lunak, khususnya aplikasi web, yang dapat menimbulkan kesalahan. Solusi yang ditawarkan berupa mengembangkan aplikasi web sebagai media informasi publikasi hasil karya mahasiswa dan menjamin kualitas perangkat lunak dengan standar ISO 25010. Aplikasi web dikembangkan dengan metode RUP dengan 4 tahap yaitu: *inception*, *elaboration*, *construction*, dan *elaboration*. Aplikasi web diuji menggunakan standar ISO 25010 dengan 5 aspek: *functional suitability*, *usability*, *performance efficiency*, *reliability*, dan *maintainability*. Sehingga penelitian ini menghasilkan produk aplikasi web yang telah teruji.

#### **D. Pertanyaan Penelitian**

1. Bagaimana mengembangkan Aplikasi Web Karyaku sebagai direktori karya mahasiswa Universitas Negeri Yogyakarta sebagai media informasi yang digunakan untuk mempublikasikan hasil karya mahasiswa secara daring dengan metode *Rational Unified Process* (RUP)?
2. Apakah Aplikasi Web Karyaku sebagai direktori karya mahasiswa Universitas Negeri Yogyakarta memenuhi aspek *functional suitability*?
3. Apakah Aplikasi Web Karyaku sebagai direktori karya mahasiswa universitas Negeri Yogyakarta memenuhi aspek *usability*?
4. Apakah Aplikasi Web Karyaku sebagai direktori karya mahasiswa Universitas Negeri Yogyakarta memenuhi aspek *reliability*?
5. Apakah Aplikasi Web Karyaku sebagai direktori karya mahasiswa Universitas Negeri Yogyakarta memenuhi aspek *performance efficiency*?

6. Apakah Aplikasi Web Karyaku sebagai direktori karya mahasiswa Universitas Negeri Yogyakarta memenuhi aspek *maintainability*?



### **BAB III**

#### **METODE PENELITIAN**

##### **A. Jenis Penelitian**

Jenis penelitian “Pengembangan Aplikasi Web Karyaku sebagai Direktori Karya Mahasiswa Universitas Negeri Yogyakarta” adalah *Research and Development* (R&D). Metode penelitian dan pengembangan merupakan metode penelitian yang digunakan untuk menghasilkan produk tertentu dan menguji keefektifan produk tersebut (Sugiyono, 2006, hal. 297). Penelitian R&D bertujuan menghasilkan produk yang mampu menyelesaikan masalah dan sesuai kebutuhan dengan suatu model pengembangan.

##### **B. Prosedur Penelitian**

###### **1. Studi Pendahuluan Permasalahan**

Studi pendahuluan permasalahan dilakukan untuk mengetahui permasalahan-permasalahan yang dan potensi-potensi yang dapat diselesaikan (Sugiyono, 2006). Studi permasalahan dilakukan dengan menyebarkan angket mengenai kepemilikan karya, jenis karya yang telah dibuat, cara dokumentasi dan publikasi, serta kebutuhan yang sesuai. Angket ini disebarkan secara daring kepada mahasiswa Universitas Negeri Yogyakarta.

###### **2. Pengembangan Aplikasi Web**

Penelitian ini menggunakan metode pengembangan *Rational Unified Process* (RUP). RUP memiliki 4 tahapan yaitu: *Inception*, *Elaboration*, *Construction*, dan *Transition*. Berikut merupakan penjelasan dari masing-masing tahapan:

a. *Inception*

Tahap *inception* memfokuskan pada pemodelan proses bisnis (*business modeling*) dan mendefinisikan kebutuhan sistem dari sistem (*requirements*) yang akan dibuat seperti yang telah dijelaskan pada **Error! Reference source not found.** mengenai fase iterasi pada RUP. Berikut adalah tahapan yang dibutuhkan pada tahap *inception*:

- 1) Membangun *business modeling* terhadap lingkungan permasalahan yang akan diselesaikan.
- 2) Memahami *requirement* atau kebutuhan sistem yang dianalisis dari bisnis model.
- 3) Melakukan penjadwalan pada proses *project management*.
- 4) Penentuan *environment* yang akan digunakan pada pengembangan aplikasi web.

Tahap *inception* dapat menggunakan teknik pengambilan data dengan kuisioner. Hasil pengambilan data dari kuisioner akan digunakan sebagai acuan analisis kebutuhan. Hasil analisis akan digunakan sebagai dasar pembuatan model bisnis dan perkiraan jadwal.

Hasil yang akan dicapai pada tahap *inception* yaitu: pendefinisain model bisnis, *requirement* fungsi-fungsi yang akan dikembangkan dalam sistem, dan penjadwalan.

b. *Elaboration*

Tahap *elaboration* fokus pada perencanaan desain arsitektur sistem yang akan dikembangkan. Pada Gambar 2, Tahap ini berfokus pada desain sistem dan

analisis (*analysis and design*) pada arsitektur yang akan dibuat. Kegiatan yang dilakukan pada tahap *elaboration* sebagai berikut:

- 1) Deskripsi arsitektur perangkat lunak yang dikembangkan dari analisis kebutuhan.
- 2) Pembuatan *use case diagram* sistem dengan aktor yang telah diidentifikasi, *class diagram*, *sequence diagram*, dan *activity diagram*.
- 3) *Mockup* aplikasi web yang akan dikembangkan.

Hasil dari tahap *elaboration* berupa desain pemodelan sistem menggunakan UML yang terdiri dari *use case diagram*, *class diagram*, *sequence diagram*, dan *activity diagram*. Tahap *elaboration* menghasilkan *mockup* aplikasi web yang akan dikembangkan.

#### c. *Construction*

Tahap *construction* fokus pada pengembangan komponen dan fitur-fitur pada perangkat lunak yang dikembangkan. Tahap ini mengimplementasikan fitur-fitur yang telah didesain pada tahap *elaboration* menjadi kode-kode program. Kegiatan yang dilakukan pada tahap *construction* yaitu:

- 1) Implementasi program menggunakan *framewrok* Laravel.
- 2) Implementasi tampilan antarmuka pada Laravel khususnya pada *template engine* Blade.
- 3) Peninjauan terhadap fungsi-fungsi yang ada dan saling berhubungan pada *unit testing*, Peninjauan integrasi fungsi-fungsi program dengan tampilan pada *integration testing*, dan peninjauan sistem secara keseluruhan pada *system testing*.

Hasil dari tahap *contruction* berupa aplikasi web yang telah jadi pada lingkungan pengembangan yang telah diuji menggunakan *unit testing*, *integration testing*, dan *system testing*.

d. *Transtiton*

Tahap *transistion* berfokus kepada proses *deployment* atau instalasi sistem agar dapat dimengerti oleh pengguna serta proses *testing*. *Deployment* dilakukan pada *production server* yang akan digunakan untuk pengguna akhir mengkases aplikasi web. *Testing* pada tahap *transition* berupa *alpha testing* dan *beta testing*.

Hasil dari tahap *transition* berupa aplikasi web yang telah dipasang pada *production server* dan dapat diakses target pengguna. Aplikasi web telah teruji dengan *alpha testing* dan *beta testing*.

3. Analisis Perangkat Lunak Aplikasi Web

Aplikasi web yang telah dikembangkan diuji kelayakannya menggunakan standar ISO 25010 dengan aspek sebagai berikut:

- a. *Functional suitability* untuk pengujian kelengkapan dan kesesuaian fitur aplikasi web yang telah dikembangkan.
- b. *Usability* untuk pengujian kemudahan penggunaan aplikasi web pada target pengguna, mahasiswa Universitas Negeri Yogyakarta.
- c. *Performance efficiency* untuk pengujian kecepatan aplikasi web memuat halaman web yang diminta.
- d. *Reliability* untuk pengujian aplikasi web dalam melakukan respon dari permintaan dalam keadaan dan beban tertentu.
- e. *Maintainability* untuk pengujian aplikasi web dalam aspek pemeliharaan dan perawatan kode aplikasi.

### **C. Sumber Data/Subjek Penelitian**

Subjek penelitian digunakan untuk menguji dua aspek yaitu *usability* dan *functional suitability*. Pengujian *functional suitability* dilakukan oleh ahli yang telah bekerja sebagai pengembang perangkat lunak atau *software developer*. Sedangkan untuk aspek *usability* menggunakan minimal 20 responden (Nielsen J. , 2012) dari mahasiswa Universitas Negeri Yogyakarta. Sedangkan untuk menguji *reliability*, *performance efficiency*, dan *maintainability* menggunakan dokumentasi perangkat lunak.

Waktu penelitian dilaksanakan pada Maret 2018 hingga Juni 2018. Tempat penelitian pengembangan produk, pengujian produk, dan revisi produk dilaksanakan di Laboratorium Program Studi Pendidikan Teknik Informatika Universitas Negeri Yogyakarta.

### **D. Variabel Penelitian**

Variabel penelitian pada Pengembangan Aplikasi Web Karyaku sebagai Direktori Karya Mahasiswa Universitas Negeri Yogyakarta adalah pengembangan aplikasi web dan pengujian kualitas perangkat lunak dengan standar ISO 25010 yaitu *functional suitability*, *usability*, *reliability*, *performance efficiency*, dan *maintainability*.

### **E. Metode Penelitian dan Alat Pengumpul Data**

Metode pengumpulan data yang digunakan pada penelitian pada Pengembangan Aplikasi Web Karyaku sebagai Direktori Karya Mahasiswa Universitas Negeri Yogyakarta sebagai berikut:

## 1. Observasi

Observasi merupakan pengamatan untuk mengumpulkan data yang ada dalam penelitian. Observasi digunakan dalam pengujian *reliability*, *performance efficiency*, dan *maintainability*. Pengujian *reliability* mengamati proses *stress testing* menggunakan WAPT. Pengujian *performance efficiency* mengamati proses pengukuran pemuatan halaman dengan menggunakan alat GTMetrix. Sedangkan pengujian *maintainability* mengamati baris-baris kode dengan menggunakan alat PHPMetrics.

## 2. Kuesioner

Kuesioner merupakan teknik pengumpulan data secara tidak langsung untuk menerima jawaban dari subjek penelitian. Kuesioner menghemat waktu, tenaga, serta biaya dan banyak informasi yang dapat dikumpulkan sekaligus (Kusumah & Dwigatama, 2009). Instrumen yang digunakan adalah angket yang akan dijawab oleh responden. Pengumpulan data menggunakan teknik ini dilakukan untuk menguji perangkat lunak pada aspek *functional suitability* dan *usability*.

## **F. Instrumen Penelitian**

Instrumen pada Pengembangan Aplikasi Web Karyaku sebagai Direktori Karya Mahasiswa Universitas Negeri Yogyakarta terdiri dari instrumen untuk pengujian perangkat lunak berdasarkan 5 aspek yang akan diujikan, yaitu *functional suitability*, *usability*, *reliability*, *performance efficiency*, dan *maintainability*.

### 1. Instrumen pengujian aspek *functional suitability*

Pengujian aspek *functional suitability* pada aplikasi web Karyaku dilakukan dengan kuesioner daftar fungsional yang dimiliki aplikasi web yang dijabarkan dari prosedur penggunaan sistem. Aspek *functional suitability* menggunakan prinsip

*black-box testing* yang berfokus pada persyaratan fungsional dengan hasil yang diharapkan (Pressman, 2014). Aspek *functional suitability* diuji oleh ahli dalam bidang pengembang perangkat lunak atau penembang web. Instrumen untuk mengukur aspek *functional suitability* terdapat pada Tabel 3.

Tabel 3. Tabel Instrumen pengujian aspek *functional suitability*.

No.	Fungsi	Hasil yang Diharapkan	Hasil	
			Ya	Tidak
Umum				
Halaman Login				
1	Login	Fungsi login untuk pengguna masuk ke dalam sistem dan mengarahkan ke halaman beranda berjalan dengan benar		
Halaman Register				
2	Register	Fungsi pendaftaran untuk pengguna dapat berjalan dengan benar		
Halaman Jelajah				
3	Jelajah	Fungsi untuk menjelajah karya secara keseluruhan dapat berjalan dengan benar		
4	Penjelahan karya berdasarkan <i>tag</i>	Fungsi untuk menjelajah karya berdasarkan <i>tag</i> karya berjalan dengan benar.		
5	Pencarian karya berdasarkan kata kunci	Fungsi untuk mencari karya berdasarkan kata kunci pencarian berjalan dengan benar.		
Halaman Karya				
6	Melihat karya	Fungsi untuk melihat karya dari menampilkan deskripsi, <i>thumbnail</i> , galeri, video dan pembuat karya berjalan dengan benar		
7	Melaporkan karya	Fungsi untuk melaporkan karya berjalan dengan benar		

No.	Fungsi	Hasil yang Diharapkan	Hasil	
			Ya	Tidak
8	Membagikan karya dengan tombol <i>share</i>	Fungsi untuk membagikan karya ke sosial media dapat berjalan dengan benar		
<b>Pengguna</b>				
Halaman Beranda				
9	Beranda	Fungsi untuk menampilkan beranda pengguna dan menampilkan karya-karya terbaru berjalan secara benar.		
Halaman tambah karya				
10	Menambah karya baru	Fungsi untuk pengguna menambahkan karya baru dapat berjalan dengan tepat.		
Halaman sunting karya				
11	Penyuntingan informasi karya	Fungsi untuk pengguna menyunting karyanya dari judul, deskripsi dan <i>tag</i> dapat berjalan dengan benar.		
12	Mengelola <i>thumbnail</i> Karya	Fungsi untuk pengguna menyunting <i>thumbnail</i> karya berjalan dengan benar.		
13	Mengelola gambar galeri karya	Fungsi untuk pengguna mengelola gambar galeri dari mengunggah dan menghapus dapat berjalan dengan benar.		
14	Mengelola video karya	Fungsi untuk pengguna mengelola video pendukung dari menambah dan menghapus dapat berjalan dengan benar.		
Halaman sunting profil				
15	Mengelola informasi dasar profil	Fungsi untuk pengguna menyunting informasi dasar dari nama, email, nim, dan prodi dapat berjalan dengan benar.		



No.	Fungsi	Hasil yang Diharapkan	Hasil	
			Ya	Tidak
16	Mengelola gambar profil	Fungsi untuk pengguna mengunggah gambar profil dapat berjalan dengan benar.		
17	Mengganti <i>password</i>	Fungsi untuk pengguna mengganti <i>password</i> dapat berjalan dengan benar.		
Navigasi				
18	Logout user	Fungsi untuk pengguna keluar dari aplikasi dapat berjalan dengan benar.		
<b>Administrator</b>				
Halaman Login				
19	Login admin	Fungsi login untuk admin ke halaman beranda admin berjalan dengan benar		
Halaman Laporan				
20	Mengelola laporan	Fungsi pengelolaan laporan dari melihat, menghapus, serta memblokir karya dan pengguna dapat berjalan dengan benar		
Halaman <i>user</i> administrator				
21	Mengelola <i>user</i> admin	Fungsi mengelola user admin dari menambah, menghapus, dan mengganti <i>password</i> user admin dapat berjalan dengan benar.		
Halaman program studi				
22	Mengelola program studi	Fungsi mengelola program studi dari menambah, menghapus, dan mengganti nama program studi dapat berjalan dengan benar.		
Halaman <i>tag</i>				
23	Mengelola <i>tag</i>	Fungsi mengelola tag dari menambah dan menghapus dapat berjalan dengan benar.		
Navigasi				

No.	Fungsi	Hasil yang Diharapkan	Hasil	
			Ya	Tidak
24	<i>Logout</i> admin	Fungsi keluar dari sistem administrator aplikasi dapat berjalan dengan benar.		

## 2. Instrumen pengujian aspek *usability*

Pengujian aspek *usability* menggunakan *USE Questionnaire* yang dibuat oleh Arnold M. Lund (2001). *Use Questionnaire* mencakup 4 aspek *usability* pada ISO 25010 yaitu *usefulness*, *ease of use*, *ease of learning*, dan *satisfaction* yang dibagi pada 30 pertanyaan. Instrumen *usability* dengan *USE Questionnaire* menggunakan skala Likert. Skala Likert yang akan digunakan memuat 5 skala yaitu Sangat Tidak Setuju (STS), Tidak Setuju (TS), Ragu (RG), Setuju (S), dan Sangat Setuju (SS). Instrumen Usability tersaji pada Tabel 4.

Tabel 4. Instrumen aspek *usability*

No.	Pertanyaan	Jawaban				
		SS	S	RG	TS	STS
Usefulness						
1	Aplikasi ini membantu saya menjadi lebih efektif					
2	Aplikasi ini membantu saya mejadi lebih produktif					
3	Aplikasi ini bermanfaat					
4	Aplikasi ini memberikan dampak yang besar terhadap tugas saya lakukan dalam hidup saya					
5	Aplikasi ini memudahkan saya dalam mencapai hal-hal yang diinginkan					
6	Aplikasi ini menghemat waktu saya ketika saya menggunakannya					
7	Aplikasi ini sesuai dengan kebutuhan saya					
8	Aplikasi ini sesuai dengan apa yang saya harapkan					
Ease of Use						
9	Aplikasi ini mudah digunakan					

No.	Pertanyaan	Jawaban				
		SS	S	RG	TS	STS
10	Aplikasi ini praktis untuk digunakan					
11	Aplikasi ini mudah dipahami					
12	Aplikasi ini hanya memerlukan langkah-langkah singkat dalam menggunakannya					
13	Aplikasi ini dapat disesuaikan dengan kebutuhan					
14	Tidak ada kesulitan dalam menggunakan aplikasi ini					
15	Saya dapat menggunakan tanpa instruksi tertulis					
16	Saya melihat aplikasi ini sudah konsisten ketika dipergunakan					
17	Baik pengguna yang jarang maupun yang rutin akan suka menggunakan aplikasi ini					
18	Saya dapat kembali dari kesalahan dengan cepat dan mudah					
19	Saya dapat menggunakan aplikasi ini dengan berhasil setiap saya gunakan					
<i>Ease of Learning</i>						
20	Saya belajar menggunakan aplikasi ini dengan cepat					
21	Saya mengingat penggunaan aplikasi ini dengan mudah					
22	Penggunaan aplikasi ini mudah dipelajari					
23	Saya mahir menggunakan aplikasi ini dengan cepat					
<i>Satisfaction</i>						
24	Saya puas dengan aplikasi ini					
25	Saya merekomendasikan aplikasi ini kepada teman-teman					
26	Aplikasi ini menyenangkan untuk digunakan					
27	Aplikasi ini bekerja sesuai dengan yang saya inginkan					
28	Aplikasi ini sangat bagus					
29	Saya merasa harus menggunakan aplikasi ini secara maksimal					
30	Aplikasi ini nyaman untuk digunakan					

### 3. Instrumen pengujian aspek *reliability*

Pengujian aspek *reliability* pada aplikasi web Karyaku dilakukan menggunakan *tools* WAPT atau *Web Application Load, Stress, and Performance Test*. WAPT dapat memberikan beban dengan menggunakan pengguna *virtual* untuk mengakses aplikasi website sehingga dapat diketahui kinerja aplikasi website ketika diberi beban.

#### 4. Instrumen pengujian aspek *performance efficiency*

Pengujian aspek *performance efficiency* pada aplikasi web Karyaku dilakukan dengan mengukur kecepatan halaman website. Pengukuran kecepatan halaman website menggunakan *tools* GTMetrix. GTMetrix menghasilkan hasil *performances scores* yang merupakan gabungan dari Google Page Speed Insight dan Yahoo YSlow, sehingga dengan GTMetrix dapat menghasilkan kecepatan halaman website dan lama waktu yang dibutuhkan untuk memuat halaman website.

#### 5. Instrumen pengujian aspek *maintainability*

Pengujian *maintainability* ada aplikasi web Karyaku dilakukan dengan cara menghitung *Maintainability Index* (MI). Perhitungan MI berdasarkan perhitungan *McCabe's Cyclomatic Complex* (CC), *Halstead's Volume* (V), dan *Lines of Code* (LOC) menggunakan *tools* PHPMetrics.

### **G. Teknik Analisis Data**

Instrumen pada Pengembangan Aplikasi Web Karyaku sebagai Direktori Karya Mahasiswa Universitas Negeri Yogyakarta terdiri dari instrumen untuk pengujian perangkat lunak berdasarkan 5 karakteristik yang akan diujikan, yaitu *functional suitability*, *usability*, *reliability*, *performance efficiency*, dan *maintainability*.

#### 1. Analisis pengujian aspek *functional suitability*

Pengujian aspek *functionality suitability* menggunakan skala Guttman sebagai skala pengukuran dalam instrumen pengujian. Skala Guttman hanya memiliki 2 jawaban berhasil dan tidak berhasil. Jawaban berhasil diberi nilai 1 dan Jawaban tidak berhasil diberi nilai 0. Perhitungan menggunakan perhitungan dari ISO 25010 untu menganalisis data hasil pengujian *functionality* dengan rumus berikut ini:

$$X = 1 - \frac{A}{B}$$

Keterangan:

$X$  = *Functional Suitability*

$A$  = Jumlah fungsi yang tidak berfungsi secara benar

$B$  = Jumlah fungsi yag dievaluasi

Rentang nilai pengujian ini adalah diantara 0 hingga 1. Hasil pengujian *functional suitability* yang baik adalah nilainya mendekati 1. Semakin mendekati 1 maka mendindikasikan banyaknya fitur yang berhasil dibuat dan diimplementasikan.

## 2. Analisis pengujian aspek *usability*

Pengujian aspek *reliability* pada aplikasi web karyaku dilakukan dengan melakukan pembagian angket *USE Questionnaire*. *USE Questionnaire* menggunakan skala Likert yang mempunyai skala 5. Pada penelitian ini, analisis kuantitatif pada skala Likert dapat diberi nilai pada Tabel 5.

Tabel 5. Analisis Kuantitatif Jawaban skala Likert menurut Sugiyono (2006).

Pilihan Jawaban	Nilai
Sangat Setuju (SS)	5
Setuju (S)	4
Ragu-ragu (RG)	3
Tidak Setuju (TS)	2
Sangat Tidak Setuju (STS)	1

Data hasil pengujian aspek *usability* yang telah dikonversi dan dianalisis dengan menghitung jawaban dari responden dengan rumus:

$$\text{Skor}_{\text{TOTAL}} = (J_{\text{SS}} \times 5) + (J_{\text{S}} \times 4) + (J_{\text{RG}} \times 3) + (J_{\text{TS}} \times 2) + (J_{\text{STS}} \times 1)$$

Keterangan:

$\text{Skor}_{\text{TOTAL}}$  : Skor total yang didapatkan dari responden

$J_{\text{SS}}$  : Jumlah responden menjawab sangat setuju (SS)

$J_{\text{S}}$  : Jumlah responden menjawab setuju (S)

$J_{\text{rg}}$  : Jumlah responden menjawab Ragu (RG)

$J_{\text{TS}}$  : Jumlah responden menjawab tidak setuju (TS)

$J_{\text{STS}}$  : Jumlah responden menjawab sangat tidak setuju (STS)

Selanjutnya menghitung persentase nilai kuantitatif dari perhitungan sebelumnya dengan menggunakan rumus:

$$\text{Persentase} = \frac{\text{Skor}_{\text{TOTAL}}}{q \times r \times 5} \times 100$$

Keterangan:

q : Jumlah pertanyaan yang ada dalam kuesioner

r : Jumlah responden yang mengisi kuesioner

Kemudian dikonversi menjadi interpretasi nilai kualitatif berskala 5 dengan tabel konversi menurut Sudaryono (2011) seperti pada Tabel 6.

Tabel 6. Tabel kriteria Interpretasi Nilai

No	Persentase	Interpretasi Nilai
1	0% - 20%	Sangat Tidak Layak
2	21% - 40%	Kurang Layak
3	41% - 60%	Cukup Layak
4	61% - 80%	Layak
5	81% - 100%	Sangat Layak

### 3. Analisis pengujian aspek *reliability*

Pengujian aspek *reliability* pada aplikasi web Karyaku dilakukan dengan melakukan *stress testing* menggunakan perangkat lunak WAPT. Hasil dari *stress testing* harus mencapai tingkat keberhasilan sebesar 95% sesuai dengan standar Telcordia (Asthana & Olivieri, 2009) .

#### 4. Analisis pengujian aspek *performance efficiency*

Pengujian aspek *performance efficiency* pada aplikasi web Karyaku dilakukan dengan mengukur kecepatan halaman website dengan menggunakan GTMetrix. Menurut Jakeb Nielsen (2010), untuk menjaga perhatian pengguna dalam website, website harus dapat dimuat dalam waktu 10 detik atau kurang. Sehingga website yang baik harus dapat memuat halaman web dalam waktu 10 detik atau kurang.

#### 5. Analisis pengujian aspek *maintainability*

Software metrics digunakan untuk mengukur *Maintainability Index*. MI merupakan alat ukur untuk menghitung sebuah angka tunggal yang digunakan untuk menunjukkan *maintainability* pada sistem (Najm, 2014). MI dapat dihitung dengan perangkat lunak PHPMetrics dengan kriteria yang tersaji pada Tabel 7.

Tabel 7. *Maintainability Index*

<b>Maintainability Index</b>	<b>Level</b>	<b>Keterangan</b>
Lebih dari 85	<i>High Maintainable</i>	Sangat mudah dirawat
66 – 85	<i>Moderate Maintainable</i>	Normal untuk dirawat
0 – 65	<i>Difficult to Maintain</i>	Sulit untuk dirawat

## **BAB IV**

### **HASIL PENELITIAN DAN PEMBAHASAN**

#### **A. Hasil Penelitian**

Penelitian yang bertujuan untuk mengembangkan aplikasi berbasis web sebagai media informasi yang digunakan untuk mempublikasikan hasil karya mahasiswa secara daring, serta menjamin tingkat kualitas perangkat lunak dengan standar ISO 25010 dilaksanakan melalui beberapa tahap pengembangan. Kegiatan pertama dalam penelitian ini adalah tahap *inception*, tahap analisis sistem yang akan dikembangkan. Tahapan meliputi proses pengolahan hasil angket dan diolah menjadi *bussines model*.

#### **1. Pengembangan Aplikasi Web**

##### **a. Tahap Inception**

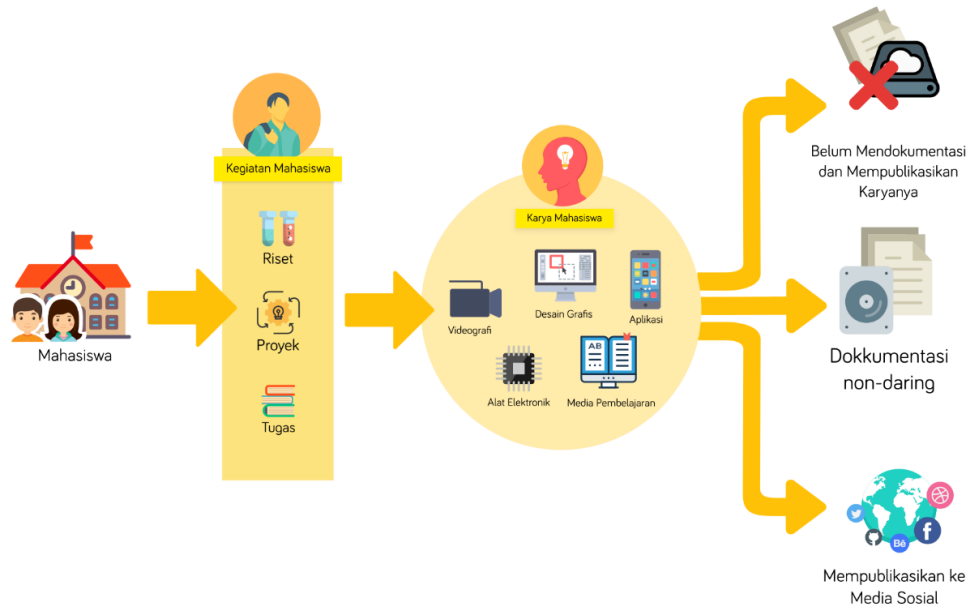
##### **1) Business Modeling**

*Business modeling* dilakukan dengan cara menyebarkan angket kepada mahasiswa-mahasiswa Universitas Negeri Yogyakarta. Hasil dari angket yang telah disebarkan telah mendapatkan 35 mahasiswa menghasilkan beberapa jawaban sebagai berikut:

- ✓ Mahasiswa memiliki banyak karya-karya yang tidak dipublikasikan.
- ✓ Mahasiswa menyimpan karya di perangkat keras dalam bentuk laporan dan tidak mempublikasikan karyanya.
- ✓ Mahasiswa membagikan karya mereka pada sosial media masing-masing seperti Facebook, Instagram, atau Github.



- ✓ Mahasiswa kesulitan mencari karya-karya mahasiswa UNY lain yang sudah pernah dibuat karena kurangnya publikasi dari mahasiswa lain.



Gambar 11. Alur Karya Mahasiswa didokumentasikan dan dipublikasikan

Selain menggunakan angket yang terlampir pada Lampiran 1, kebutuhan fungsional dapat diambil dari studi pustaka mengenai portfolio elektronik khususnya *showcase e-portfolio*, dokumen karya, dan mengenai perangkat pengembangan yang akan digunakan seperti *framework* Laravel, diagram UML, dan sebagainya.

*Business modeling* yang telah dilakukan dapat menghasilkan deskripsi aplikasi sebagai berikut:

Aplikasi ini bertujuan membantu mahasiswa mempublikasikan dan mendokumentasikan hasil karya mahasiswa. Mahasiswa dapat menambah karya yang telah dibuat, menambahkan video dan gambar yang mendukung karya. Mahasiswa bisa menjelajah karya-karya yang telah dibuat mahasiswa lain dan bisa

menjelajah berdasarkan tag yang telah ditambahkan. Mahasiswa juga dapat membagikan karya ke media sosial yang lain.

## **2) Requirements**

Aktivitas *requirement* merupakan tahap analisis kebutuhan fungsionalitas dari aplikasi. Analisis kebutuhan fungsional dari aplikasi didasarkan pada hasil angket dan studi literatur. Kebutuhan fungsi pada pengembangan aplikasi web Karyaku sebagai direktori karya mahasiswa adalah sebagai berikut:

### **a) Analisis Fungsionalitas**

Kebutuhan fungsionalitas sistem dapat diambil dari angket yang telah dibagikan. Hasil kebutuhan fungsional yang telah didapat adalah sebagai berikut:

- ✓ Aplikasi dapat mengunggah karya, serta tiap karya dapat mencantumkan deskripsi, gambar, dan video pendukung.
- ✓ Aplikasi mampu melakukan pencarian karya-karya yang sudah ada.
- ✓ Aplikasi memiliki fitur membagikan karya yang telah diunggah oleh mahasiswa.
- ✓ Aplikasi memiliki fitur klasifikasi karya berdasarkan kategori yang telah dimasukan oleh pengguna.
- ✓ Aplikasi memiliki fitur pelaporan atau klarifikasi yang jelas.

## **3) Project Management**

Pada tahap *inception*, peneliti melakukan penjadwalan dalam pembuatan produk. Tujuan dari penjadwalan ini adalah sebagai pedoman bagi pengembang agar penelitian dapat berjalan dengan efektif dan sesuai dengan estimasi waktu yang telah ditetapkan. Penjadwakan pengerjaan aplikasi web karyaku tersaji pada Tabel 8.

Tabel 8. Penjadwalan pengerjaan aplikasi web

No.	Nama Kegiatan	Durasi	Mulai	Selesai
1	Inception	2 Minggu	05/03/2018	18/03/2018
2	Elaboration	1 Minggu	19/03/2018	25/03/2018
3	Construction	6 Minggu	26/03/2018	29/04/2018
4	Transition	2 Minggu	30/04/2018	15/05/2018

Berdasarkan tabel penjadwalan pengerjaan aplikasi web, aplikasi web Karyaku sebagai direktori karya mahasiswa ini dikembangkan selama 11 minggu yakni dari 5 Maret 2018 sampai 15 Mei 2018.

#### 4) Environment

Perangkat keras (*hardware*) yang dibutuhkan untuk mengembangkan aplikasi web direktori karya mahasiswa adalah menggunakan satu unit laptop dengan spesifikasi yang tercantum pada Tabel 9.

Tabel 9. Spesifikasi laptop yang digunakan untuk mengembangkan aplikasi.

No	Spesifikasi
1	Intel Core i5-5200
2	RAM 4096 MB
3	HDD 500 GB
4	Nvidia GeForce 930M
5	Operating System Windows 8.1

Sedangkan perangkat lunak (*software*) yang dibutuhkan untuk mengembangkan aplikasi web karyaku adalah sebagai berikut:

- Laragon 32-bit sebagai *web server*.
- Laravel yang digunakan sebagai *framework* untuk membangun aplikasi web.
- Sublime Text 3 Sebagai *text editor* untuk membangun website.
- Pencil digunakan untuk membuat *wireframe* halaman website.

- e) StarUML digunakan untuk membuat desain UML.
- f) *Web Browser* Mozilla Firefox.

## **b. Tahap Elaboration**

Tahap *elaboration* merupakan tahap yang berfokus pada perancangan arsitektur aplikasi menggunakan diagram UML seperti *use case diagram*, *class diagram*, *activity diagram*, dan *sequence diagram*, serta perancangan tampilan antarmuka aplikasi.

### **1) Business Modeling**

*Business Modeling* pada tahap *elaboration* fokus kepada fitur tambahan yang diperlukan. Pada aplikasi ini dibutuhkan administrator untuk melakukan pengelolaan data-data yang ada pada aplikasi seperti *tag*, laporan, program studi, dan moderasi hasil laporan pengguna.

Dari penambahan *business modeling*, maka dapat didapatkan deskripsi aplikasi sebagai berikut:

Aplikasi ini bertujuan membantu mahasiswa mempublikasikan dan mendokumentasikan hasil karya mahasiswa. Mahasiswa dapat menambah karya yang telah dibuat, menambahkan video dan gambar yang mendukung karya. Mahasiswa bisa menjelajah karya-karya yang telah dibuat mahasiswa lain dan bisa menjelajah berdasarkan tag yang telah ditambahkan. Mahasiswa juga dapat membagikan karya ke media sosial yang lain. Administrator dapat mengelola tags, mengelola program studi, dan mengelola administrator itu sendiri. Administrator juga dapat mengelola laporan yang masuk dari pengguna dan memoderasi karya dan pengguna berdasarkan laporan yang masuk.

## 2) Requirement

*Requirement analysis* pada tahap *elaboration* terdapat penambahan fungsionalitas yang perlu dikembangkan. Kebutuhan fungsionalitas yang ditambahkan adalah sebagai berikut:

- ✓ Administrator aplikasi dapat menambahkan dan menghapus pilihan *tags* pada karya.
- ✓ Administrator aplikasi dapat mengelola (menambah, menyunting, dan menghapus) data program studi
- ✓ Administrator dapat mengelola (menambah, menghapus, dan mengganti *password*) pengguna administrator
- ✓ Administrator dapat memoderasi (menghapus atau mengembalikan) karya dan pengguna (mahasiswa) berdasarkan laporan yang telah masuk

## 3) Analysis and Design

### a) Use Case Diagram

#### (1) Definisi Aktor

Aktor-aktor yang terlibat dalam aplikasi web Karyaku sebagai direktori karya mahasiswa tersaji pada Tabel 10.

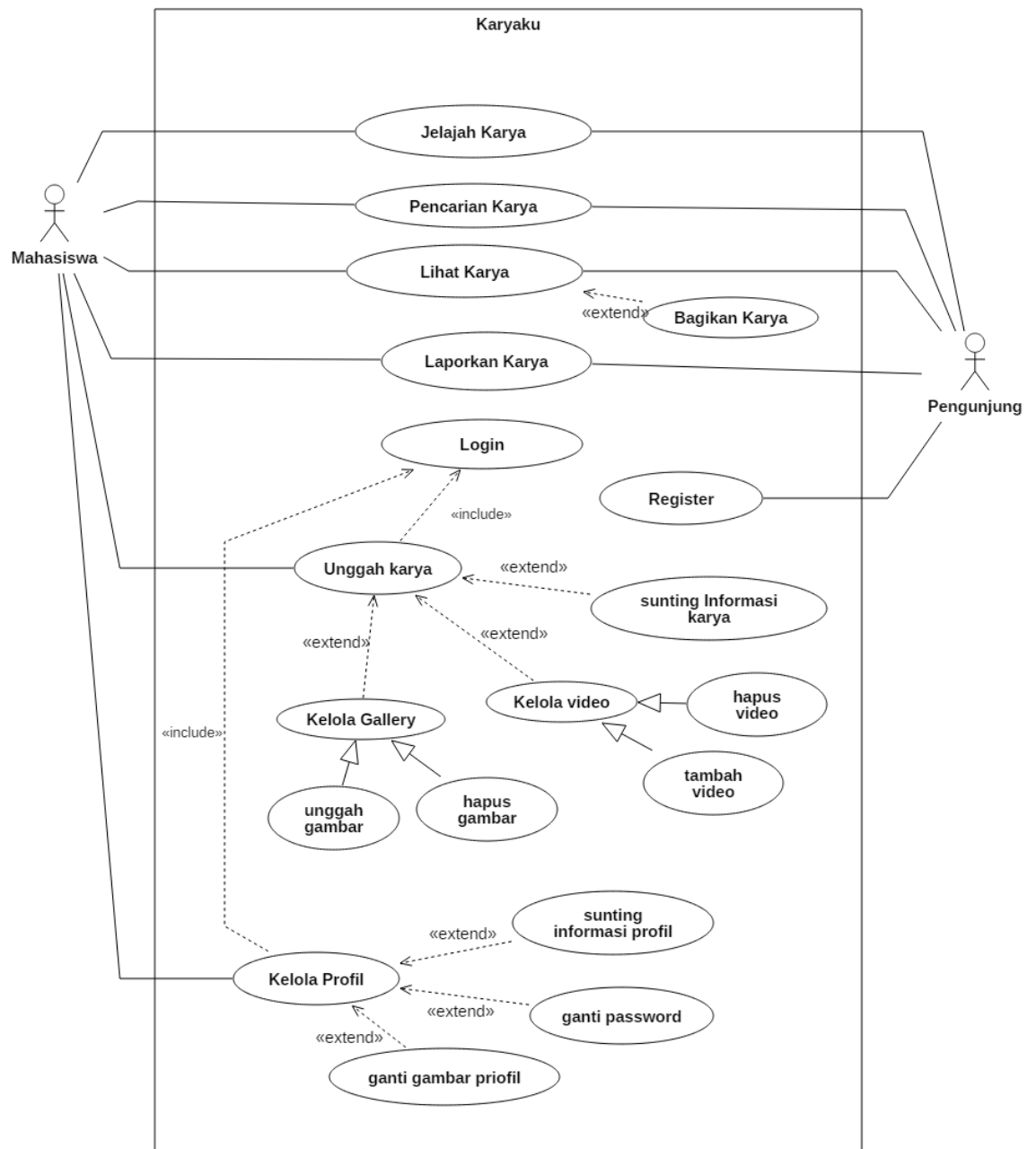
Tabel 10. Definisi Aktor pada *Use Case Diagram*

No	Aktor	Deskripsi
1	Pengunjung	Pengunjung adalah pengguna yang tidak memiliki autentifikasi ke dalam sistem tetapi bisa melakukan beberapa proses yang diizinkan tanpa melakukan autentifikasi.

2	Mahasiswa	Mahasiswa adalah pengguna yang dapat melakukan penambahan karya dan mengelola karya (mengelola galeri, mengelola video, dan mengelola informasi dasar) yang telah diunggah.
3	Admin	Admin adalah pihak yang berwenang untuk mengelola <i>tag</i> , menerima laporan terkait karya-karya yang telah diunggah, serta melakukan moderasi karya dan pengguna.

## (2) *Use Case Diagram* Mahasiswa dan Pengunjung

Aktor dalam Gambar 12 adalah mahasiswa dan pengunjung. Hak akses yang dimiliki oleh pengunjung antara lain: melakukan penjelajahan karya, mencari karya, melihat karya, melakukan registrasi, dan melaporkan karya. Hak akses yang dimiliki mahasiswa antara lain: melakukan penjelajahan karya, mencari karya, melihat karya, melakukan registrasi, melaporkan karya, mengunggah karya, dan mengelola profil. Deskripsi untuk masing-masing *use case* pada gambar diatas tersaji dalam Tabel 11 dan skenario *use case* terdapat pada Lampiran 5.



Gambar 12. Use Case Diagram Mahasiswa dan Pengunjung

Tabel 11. Deskripsi *Use Case Diagram* Mahasiswa dan Pengunjung

No	Nama <i>Use Case</i>	Deskripsi
1	Jelajah Karya	<i>Use case</i> untuk melakukan penjelajahan karya secara keseluruhan atau berdasarkan <i>tag</i>
2	Pencarian Karya	<i>Use case</i> untuk melakukan pencarian karya berdasarkan kata kunci yang telah dimasukkan oleh pengguna.
3	Lihat Karya	<i>Use case</i> untuk melihat karya
a	Bagikan Karya	<i>Use case</i> untuk membagikan karya oleh pengguna
4	Laporan Karya	<i>Use case</i> untuk melaporkan karya oleh pengguna
5	Unggah Karya	<i>Use case</i> untuk mengunggah karya oleh mahasiswa
6	Sunting Informasi	<i>Use case</i> untuk menyunting karya oleh mahasiswa
7	Kelola Galeri	<i>Use case</i> untuk mengelola galeri karya oleh mahasiswa
a	Unggah Gambar	<i>Use case</i> untuk mengunggah gambar mendukung galeri karya oleh mahasiswa
b	Hapus Gambar	<i>Use case</i> untuk menghapus gambar mendukung karya oleh mahasiswa
8	Kelola Video	<i>Use case</i> untuk mengelola video pendukung karya oleh mahasiswa
a	Tambah Video	<i>Use case</i> untuk menambahkan video pendukung karya oleh mahasiswa
b	Hapus Video	<i>Use case</i> untuk menghapus video pendukung karya oleh mahasiswa
9	Kelola Profil	<i>Use case</i> untuk mengelola kelengkapan informasi profil mahasiswa



No	Nama <i>Use Case</i>	Deskripsi
a	Sunting Informasi Profil	<i>Use case</i> untuk mengelola informasi dasar profil mahasiswa
b	Ganti Gambar Profil	<i>Use case</i> untuk mengelola gambar profil mahasiswa
c	Ganti <i>Password</i>	<i>Use case</i> untuk mengganti <i>password</i> akun profil mahasiswa
12	Login	<i>Use case</i> untuk masuk kedalam aplikasi sebagai mahasiswa
13	Register	<i>Use case</i> untuk mendaftar akun sebagai mahasiswa

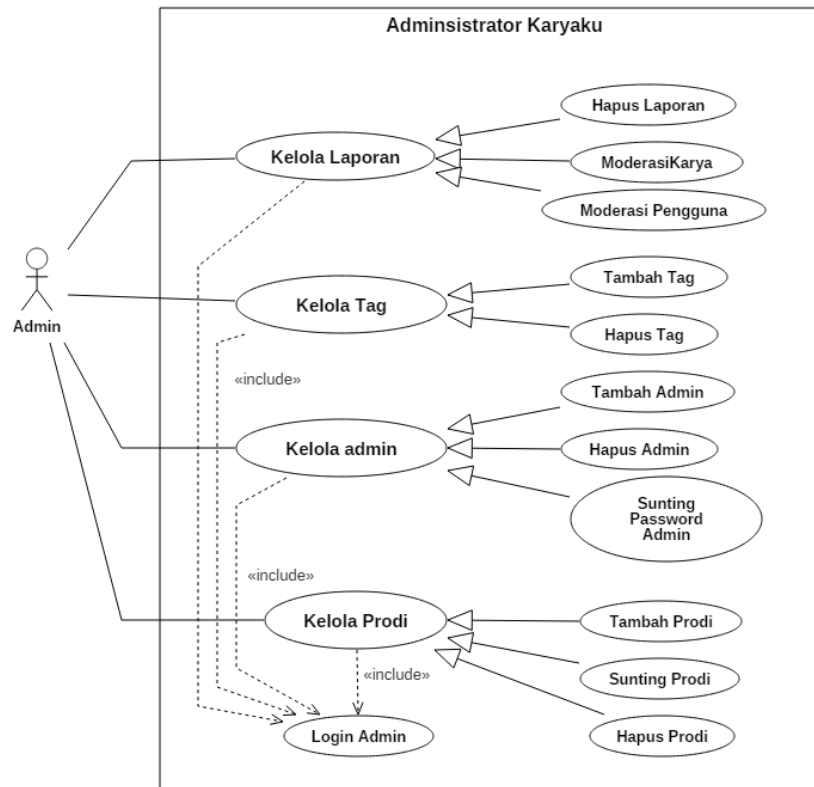
Setelah penentuan *use case* mahasiswa dan pengunjung, langkah selanjutnya mendefinisikan skenario *use case* untuk mahasiswa dan pengunjung aplikasi. Skenario *use case* digunakan untuk menjabarkan kegiatan yang akan dilakukan oleh pengguna sistem. Skenario *use case* Unggah Karya terdapat pada Tabel 12 dan skenario *use case* yang lainnya terdapat pada Lampiran 5.

Tabel 12. Skenario *Use Case* Unggah Karya

<b>Nama Use Case</b>	Unggah Karya
<b>Aktor</b>	Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk mengunggah gambar mendukung galeri karya oleh mahasiswa
<b>Pre Condition</b>	Aktor mahasiswa terautentikasi
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Mahasiswa membuka <i>form</i> tambah karya</li> <li>2. Sistem menampilkan <i>form</i></li> <li>3. Mahasiswa mengisi <i>form</i> dan mengeklik simpan</li> <li>4. Sistem memvalidasi input</li> <li>5. Sistem menyimpan ke dalam basis data</li> <li>6. Sistem menampilkan pesan sukses</li> </ol>
<b>Exception</b>	4a. Sistem memberikan kegagalan validasi

	4a. 1. Sistem mengembalikan ke form 4a. 2. Pengguna kembali ke proses 3
<b>Post Condition</b>	Mahasiswa memiliki karya yang telah diunggah

### (3) Use Case Diagram Admin



Gambar 13. Use Case Diagram Administrator

Aktor dalam Gambar 13 diatas adalah mahasiswa dan pengunjung. Hak akses yang dimiliki oleh administrator antara lain: mengelola *tag*, mengelola prodi, mengelola administrator, serta memoderasi karya dan user dari laporan. Deskripsi untuk masing-masing use case pada gambar diatas tersaji dalam Tabel 13.

Tabel 13. Deskripsi *Use Case Diagram* Admin

No	Nama <i>Use Case</i>	Deskripsi
1	Kelola <i>Tag</i>	<i>Use case</i> untuk melakukan pengelolaan <i>tag</i> karya yang akan disediakan oleh aplikasi

No	Nama <i>Use Case</i>	Deskripsi
a	Tambah <i>Tag</i>	<i>Use case</i> untuk melakukan penambahan <i>tag</i> karya yang akan disediakan oleh aplikasi
b	Hapus <i>Tag</i>	<i>Use case</i> untuk melakukan penghapusan <i>tag</i> karya yang akan disediakan oleh aplikasi
2	Kelola Administrator	<i>Use case</i> untuk melakukan pengelolaan akun administrator aplikasi
a	Tambah Admin	<i>Use case</i> untuk melakukan akun administrator aplikasi
b	Hapus Admin	<i>Use case</i> untuk melakukan penghapusan akun administrator aplikasi
c	Sunting Password Admin	<i>Use case</i> untuk melakukan penggantian <i>password</i> akun administrator aplikasi
3	Kelola Prodi	<i>Use case</i> untuk melakukan pengelolaan data program studi yang disediakan oleh aplikasi
a	Tambah Prodi	<i>Use case</i> untuk melakukan penambahan data program studi yang disediakan oleh aplikasi
b	Sunting Prodi	<i>Use case</i> untuk melakukan penyuntingan data program studi yang disediakan oleh aplikasi
c	Hapus Prodi	<i>Use case</i> untuk melakukan penghapusan data program studi yang disediakan oleh aplikasi
4	Kelola Laporan	<i>Use case</i> untuk melakukan pengelolaan laporan yang masuk dari pengguna
a	Hapus Laporan	<i>Use case</i> untuk melakukan penghapusan laporan yang masuk dari pengguna
b	Moderasi Karya	<i>Use case</i> untuk melakukan pemblokiran dan pengembalian karya berdasarkan laporan yang telah masuk dari pengguna
c	Moderasi Pengguna	<i>Use case</i> untuk melakukan pemblokiran dan pengembalian akun pengguna (mahasiswa)

No	Nama <i>Use Case</i>	Deskripsi
		berdasarkan laporan yang telah masuk dari pengguna
5	Login Admin	<i>Use case</i> untuk melakukan login ke dalam akun administrator aplikasi

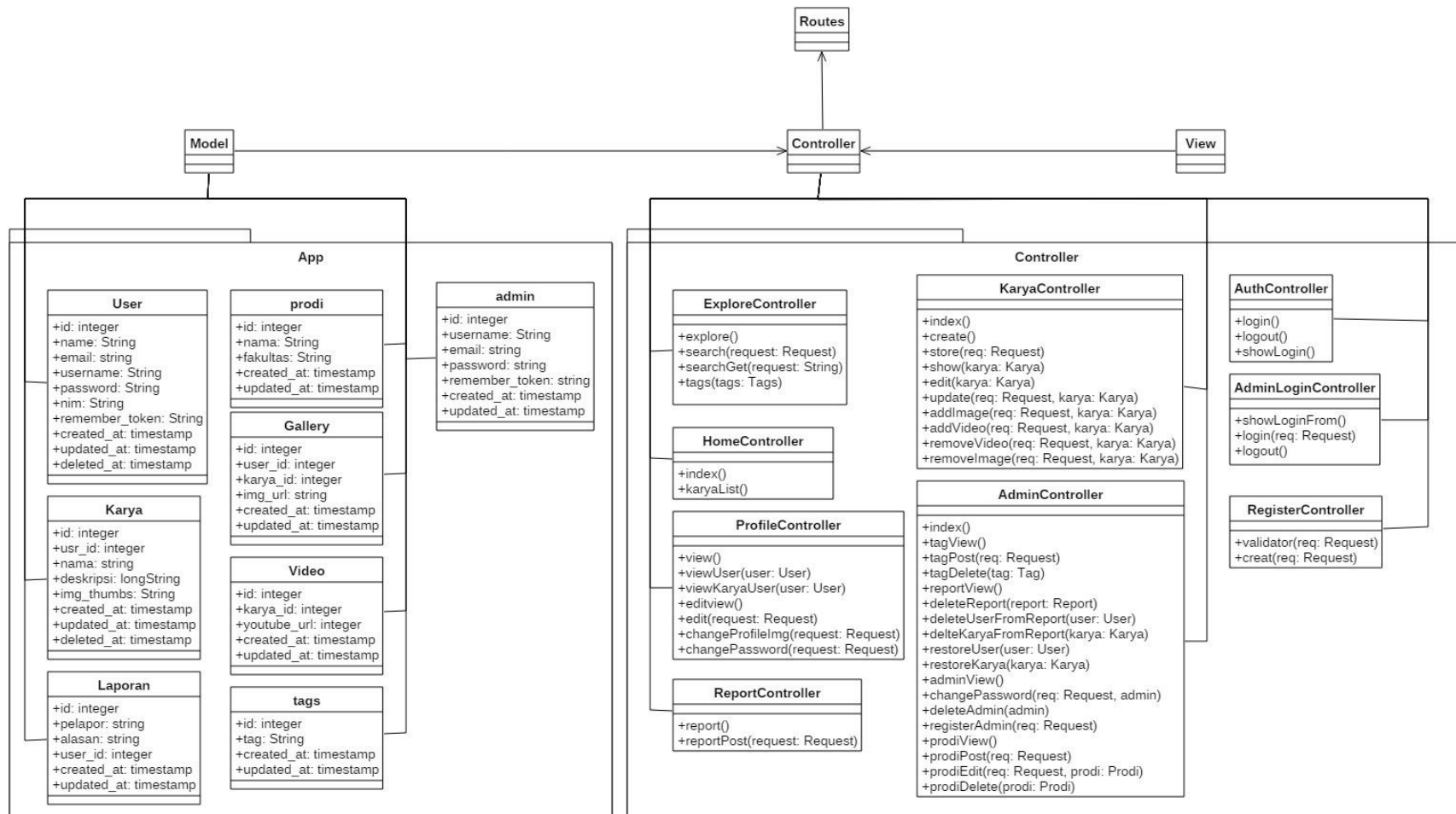
Setelah penentuan *use case* administrator, langkah selanjutnya mendefinisikan skenario *use case* untuk administrator. Skenario *use case* digunakan untuk menjabarkan kegiatan yang akan dilakukan oleh administrator dan sistem. Skenario *use case* Moderasi Karya terdapat pada Tabel 14 dan skenario *use case* lain terlampir pada Lampiran 5.

Tabel 14. Skenario *Use Case* Moderasi Karya

<b>Nama Use Case</b>	Moderasi Karya
<b>Aktor</b>	Administrator
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan pemblokiran dan pengembalian karya berdasarkan laporan yang telah masuk dari pengguna
<b>Pre Condition</b>	1. Administrator sudah terautentifikasi 2. Laporan masuk mengenai karya
<b>Main Flow</b>	1. Administrator memilih karya yang akan dimoderasi. 2. Administrator memilih blokir karya jika karya atau pengguna belum terblokir. 3. Sistem memblokir pengguna.
<b>Exception</b>	2a. Administrator memilih kembalikan pengguna jika karya atau pengguna terblokir. 2a. 1. Sistem mengembalikan karya.
<b>Post Condition</b>	Karya mahasiswa terblokir atau kembali

*b) Class Diagram*

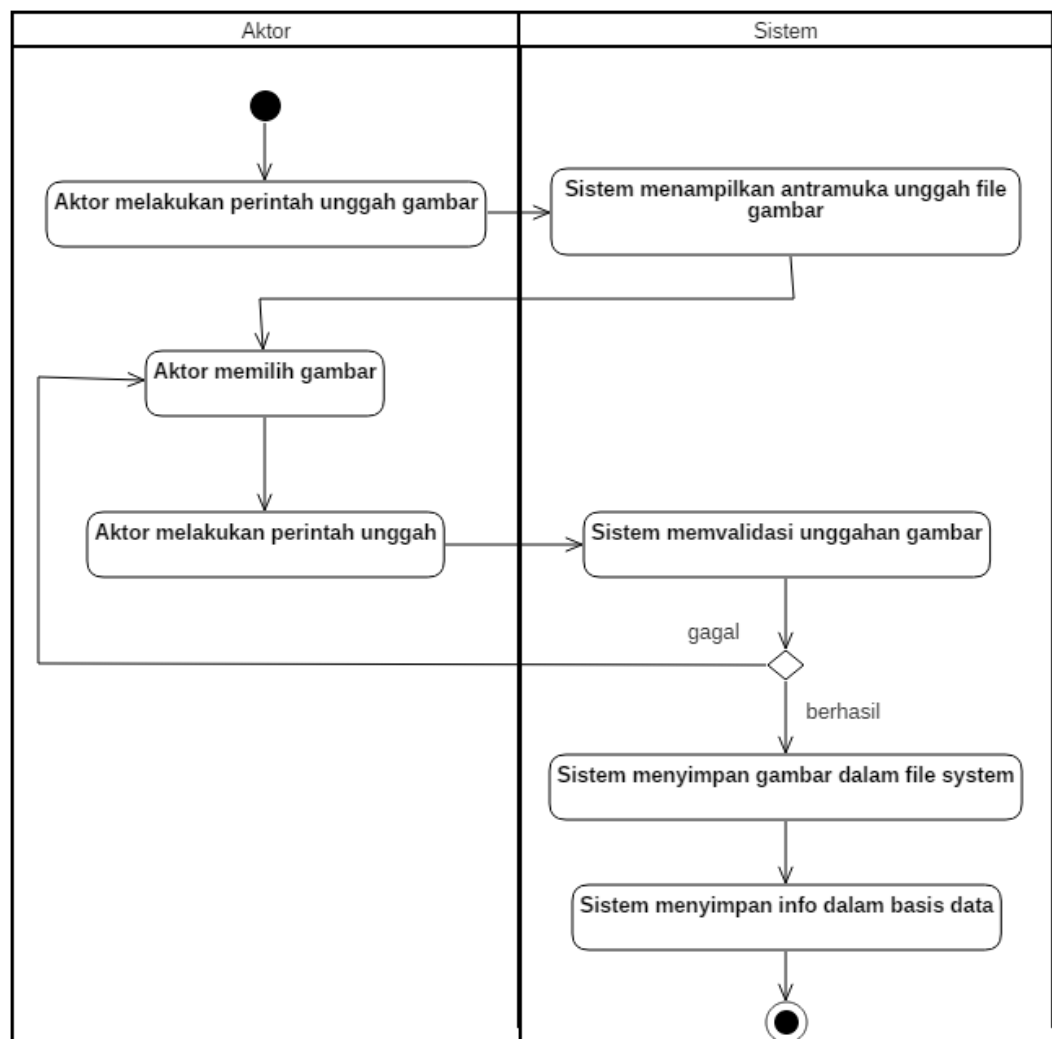
Desain *class diagram* pada aplikasi ini mengacu konsep *framework* Laravel yang berarsitektur MVC (*Model View Controller*). Desain *class diagram* memisahkan antara model dan controller. Desain diagram dapat dilihat pada Gambar 14.



Gambar 14. *Class Diagram* Aplikasi

c) *Activity Diagram*

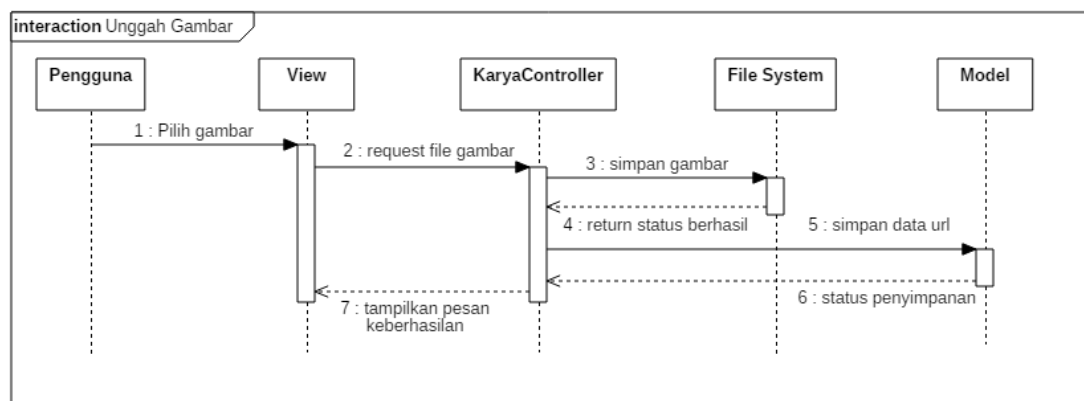
Berdasarkan *use case diagram* diatas, dihasilkan *Activity Diagram*. *Activity diagram* bertujuan untuk menggambarkan proses interaksi antara aktor dan sistem. Salah satu *activity diagram* yang dibuat yaitu aktivitas unggah karya tersaji pada Gambar 15. Sedangkan untuk daftar *activity diagram* terdapat pada Lampiran 6.



Gambar 15. *Activity Diagram* Unggah Karya

d) *Sequence Diagram*

Dari *use case diagram* yang telah dibuat, dapat dibuat pula *sequence diagram*. *Sequence diagram* yang dibuat adalah *sequence* untuk aktivitas unggah karya yang tersaji pada Gambar 16. Sedangkan untuk daftar *activity diagram* terdapat pada Lampiran 7.



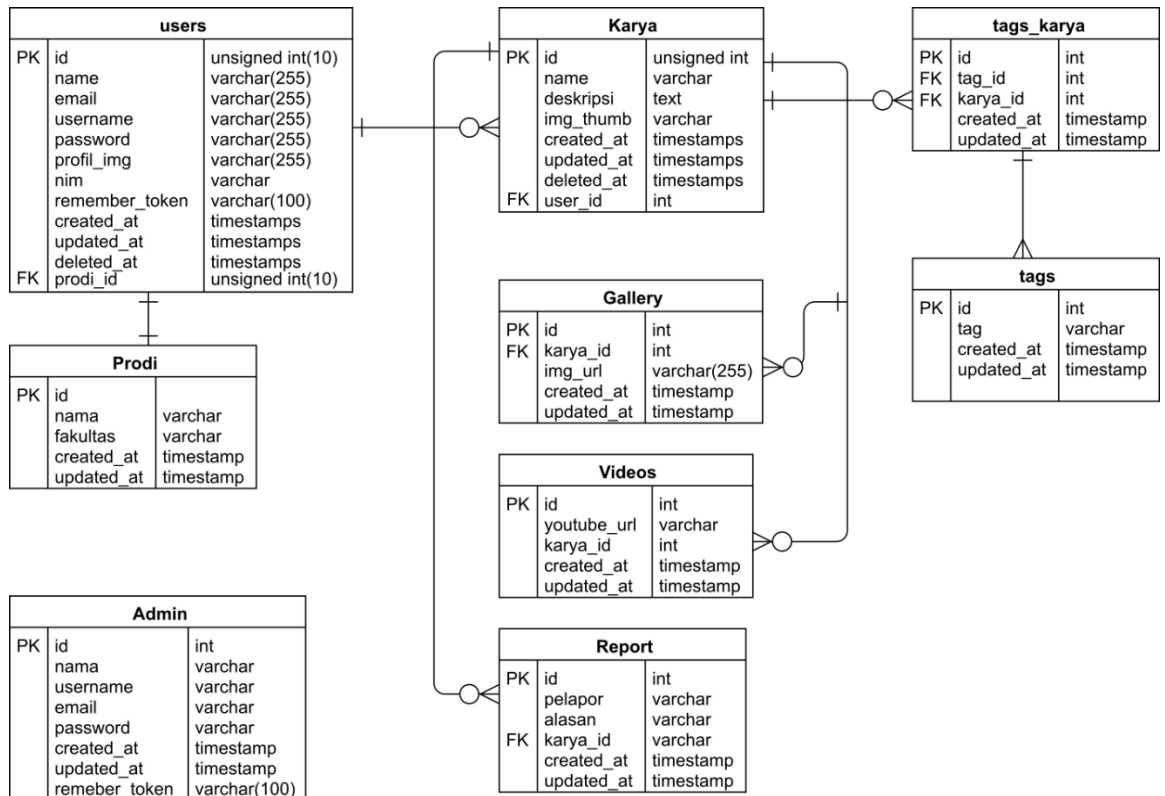
Gambar 16. *Sequence Diagram* Unggah Karya

e) *Desain Basis Data*

Tahap *elaboration* juga mencakup kegiatan desain basis data. Kegiatan ini bertujuan menghasilkan desain tabel dan relasinya yang akan diimplementasikan pada basis data MySQL.

Desain database mengacu pada model yang telah didesain pada *Class Diagram* yang tersaji pada Gambar 14. Desain basis data menggunakan pemodelan ERD (*Entity Relationship Diagram*) dengan notasi *crow's foot*. Desain basis data tersaji pada Gambar 17 dan deskripsi tabel terdapat pada Tabel 15.





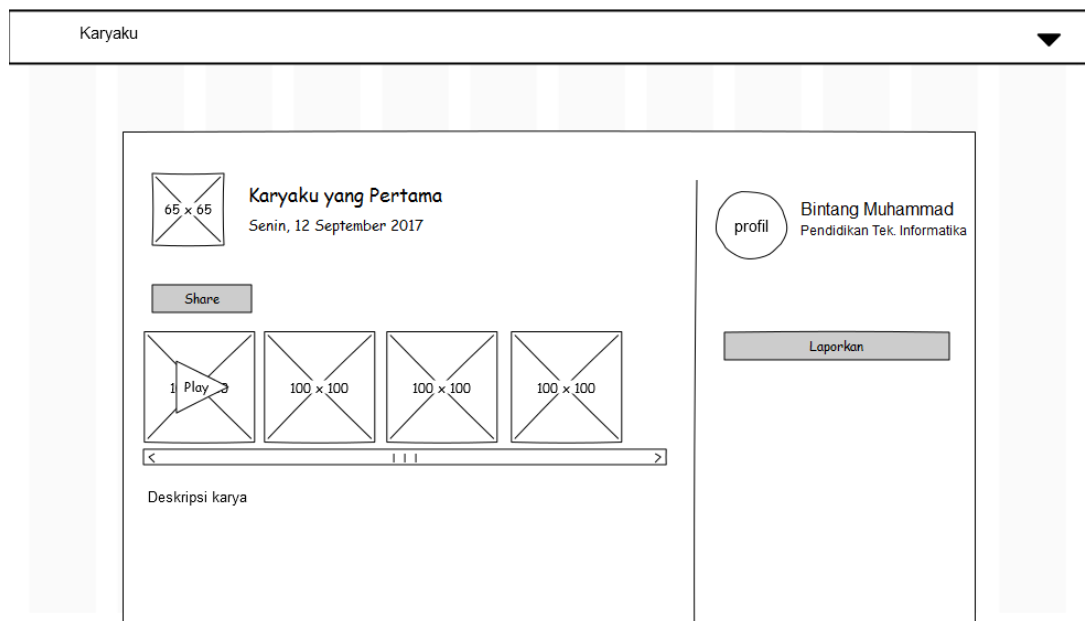
Gambar 17. Desain ERD Basis Data Aplikasi

Tabel 15. Tabel Deskripsi Desain Basis Data

No	Nama Tabel	Deskripsi
1	Users	Menyimpan data akun mahasiswa
2	Prodi	Menyimpan data program studi
3	Karya	Menyimpan data karya mahasiswa
4	Galery	Menyimpan data galeri gambar karya
5	Videos	Menyimpan data tautan video karya
6	Report	Menyimpan data laporan masuk
7	Tags	Menyimpan data <i>tag</i> untuk karya
8	Tags_karya	Menyimpan data <i>tag</i> dan karya
9	Admin	Menyimpan data akun administrator

#### 4) Implementation

Kegiatan *implementation* dalam tahap *elaboration* berupa desain tampilan pengguna. Kegiatan desain tampilan pengguna menggunakan perangkat lunak Pencil. Desain tampilan yang dibuat masih berupa *wireframe* atau gambaran kasar tampilan. Salah satu desain tampilan aplikasi yang telah digambar adalah tampilan lihat karya yang dapat dilihat pada Gambar 18.



Gambar 18. Desain Tampilan Lihat Karya

Desain tampilan aplikasi Karyaku yang lainnya dapat dilihat pada Lampiran 9. Daftar desain tampilan aplikasi Karyaku sebagai direktori karya mahasiswa dapat dilihat pada Tabel 16.

Tabel 16. Daftar Desain Tampilan Aplikasi

No.	Tampilan Aplikasi
<b>Pengguna</b>	
1	Halaman Register
2	Halaman Beranda
3	Halaman Unggah Karya
4	Halaman Pencarian
5	Halaman Jelajah
6	Halaman Karya
7	Halaman Sunting Karya
8	Halaman Pelaporan Karya
9	Halaman Sunting Profil
10	Halaman Profil
<b>Administrator</b>	
11	Halaman Login
12	Halaman <i>Dashboard</i>
13	Halaman Kelola Laporan
14	Halaman Kelola <i>Tags</i>
15	Halaman Kelola Prodi
16	Halaman Kelola Administrator

### c. Tahap Construction

Tahapan *construction* berfokus pada implementasi dari desain-desain sistem yang telah dibuat menjadi kode-kode program.

#### 1) Implementation

Implementasi pada tahap *construction* dibagi menjadi 2 yaitu implementasi program dan tampilan.

## a) Implementasi Program

Implementasi program pada aplikasi web Karyaku menggunakan *framework* Laravel 5.5 dan Sublime Text 3 sebagai *text editor* serta menggunakan Laragon sebagai *web server* untuk lingkungan pengembangan. Berikut ini merupakan potongan kode yang terdiri dari *routes*, *controller*, *model*, dan *view*.

### (1) Potongan Kode *Routes*

Kode routes digunakan untuk membuat akses URL dan menghubungkannya ke *View* dan *Controller* pada Laravel. Potongan kode *routes* dapat dilihat pada Gambar 19.



```
Route::group(['prefix' => 'karya', 'as' => 'karya.'], function () {  
    // CRUD karya user  
    Route::get('create/', 'KaryaController@create')->name('buatview')->middleware('auth');  
    Route::post('create/', 'KaryaController@store')->name('buat')->middleware('auth');  
    Route::post('{karya}/thumbs/', 'KaryaController@addThumbs')->name('buat.thumbs')->middleware('auth');  
    Route::post('{karya}/img/upload/', 'KaryaController@addImage')->name('buat.gambar')->middleware('auth');  
    Route::post('{karya}/video/', 'KaryaController@addVideo')->name('buat.video')->middleware('auth');  
    Route::post('{karya}/img/delete', 'KaryaController@removeImage')->name('hapus.gambar')->middleware('auth');  
    Route::post('{karya}/video/delete', 'KaryaController@removeVideo')->name('hapus.video')->middleware('auth');  
    Route::get('{karya}/edit', 'KaryaController@edit')->name('editview')->middleware('auth');  
    Route::post('{karya}/edit', 'KaryaController@update')->name('edit')->middleware('auth');  
  
    Route::get('{karya}', 'KaryaController@show')->name('tampil');  
});
```

Gambar 19. Potongan kode *routes*

### (2) Potongan Kode *Model*

Kode model digunakan untuk melakukan operasi-operasi manipulasi data pada basis data. Salah satu contoh *model* yaitu *model* Karya ditunjukkan pada Gambar 20.

```

class Karya extends Model
{
    use SoftDeletes;

    protected $table = "karya";

    public function gallery() {
        return $this->hasMany('App\Gallery', 'karya_id');
    }

    public function videos() {
        return $this->hasMany('App\Video', 'karya_id');
    }

    public function user() {
        return $this->belongsTo('App\User', 'user_id')->withTrashed();
    }

    public function tags() {
        return $this->belongsToMany('App\Tags', 'tags_karya');
    }
}

```

Gambar 20. Potongan kode *model*

### (3) Potongan Kode *Controller*

Kode *controller* digunakan untuk melakukan proses-proses logika yang telah masuk ke aplikasi. Salah satu potongan kode dari *controller* karya ditunjukkan pada Gambar 21.

```

class KaryaController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        // return Tags::all();
        $karya = Karya::select('nama', 'id')->with('tags')->get();
        return $karya;
    }
}

```

Gambar 21. Potongan kode *Controller*

#### (4) Potongan Kode *View*

Kode *view* digunakan untuk menangani antarmuka dari aplikasi. Kode *view* pada Laravel menggunakan *template engine* Blade yang memungkinkan penggunaan *template*. Potongan kode *view layout* utama pada aplikasi web Karyaku terdapat pada Gambar 22.



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.0/css/bootstrap.min.css">
  <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.0.10/css/all.css"
  integrity="sha384--d0P83n9kaQMCwj8F4RJB66tzIwOKmrdb46+porD/OvrJ+37WqIM7UoBtwH06Nlg" crossorigin="anonymous">

  <link rel="stylesheet" href="{{ asset('/css/karyaku.css') }}">
  @stack('css')
  <link rel="shortcut icon" href="{{ asset('/img/favicon.png') }}" type="image/x-icon">
  <meta name="theme-color" content="#0093E9">
  <title>@yield('title')Karyaku</title>
</head>
<body>

  @yield('content')

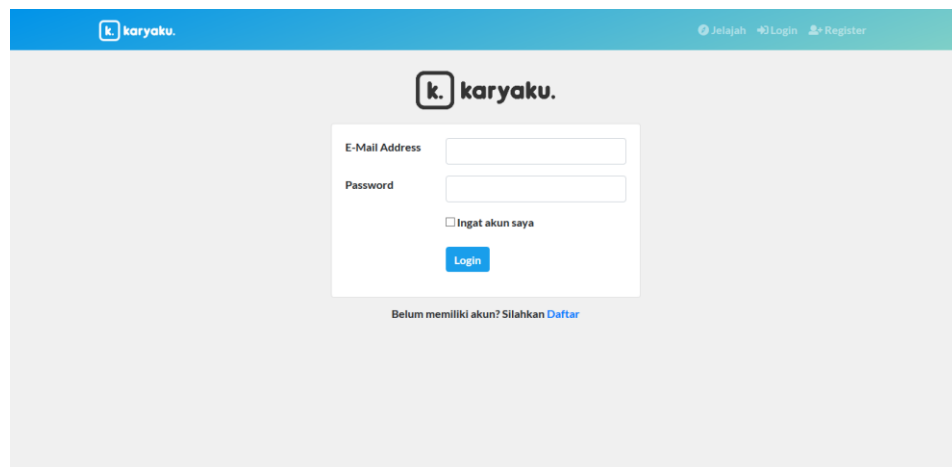
  <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.0/js/bootstrap.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.5.16/vue.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/axios/0.18.0/axios.min.js"></script>
  @stack('js')
</body>
</html>
```

Gambar 22. Potongan kode *view*

#### b) Implementasi tampilan

Pengimplementasian *user interface* dari *wireframe* tahap *elaboration* dilakukan pada tahap *construction*. Hasil implementasi yang telah dibuat dapat dilihat pada Gambar 23, Gambar 24, Gambar 25, Gambar 26, Gambar 27, Gambar 28, Gambar 29, Gambar 30, Gambar 31, Gambar 32, dan Gambar 33. Hasil implementasi yang lainnya dapat dilihat pada Lampiran 9.

## (1) Halaman Login

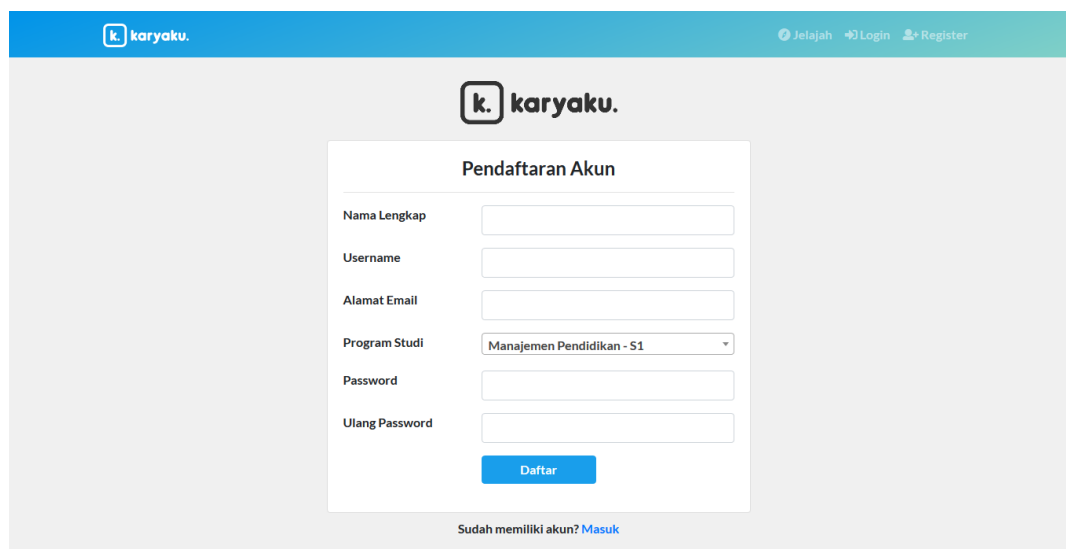


The screenshot shows the login page of the Karyaku application. At the top, there is a blue header with the Karyaku logo on the left and navigation links 'Jelajah', 'Login', and 'Register' on the right. The main content area has a light gray background. In the center, there is a white login form with the Karyaku logo at the top. The form contains two input fields: 'E-Mail Address' and 'Password'. Below these fields is a checkbox labeled 'Ingat akun saya' (Remember my account). A blue 'Login' button is positioned below the checkbox. At the bottom of the form, there is a link that says 'Belum memiliki akun? Silahkan [Daftar](#)' (Don't have an account? Please [Register](#)).

Gambar 23. Implementasi Tampilan Halaman Login

Untuk masuk akun ke aplikasi, login harus menggunakan *email* dan *password* sebagai kredensial untuk masuk aplikasi. Sehingga halaman login memuat 2 masukan *email* dan *password*.

## (2) Halaman Register

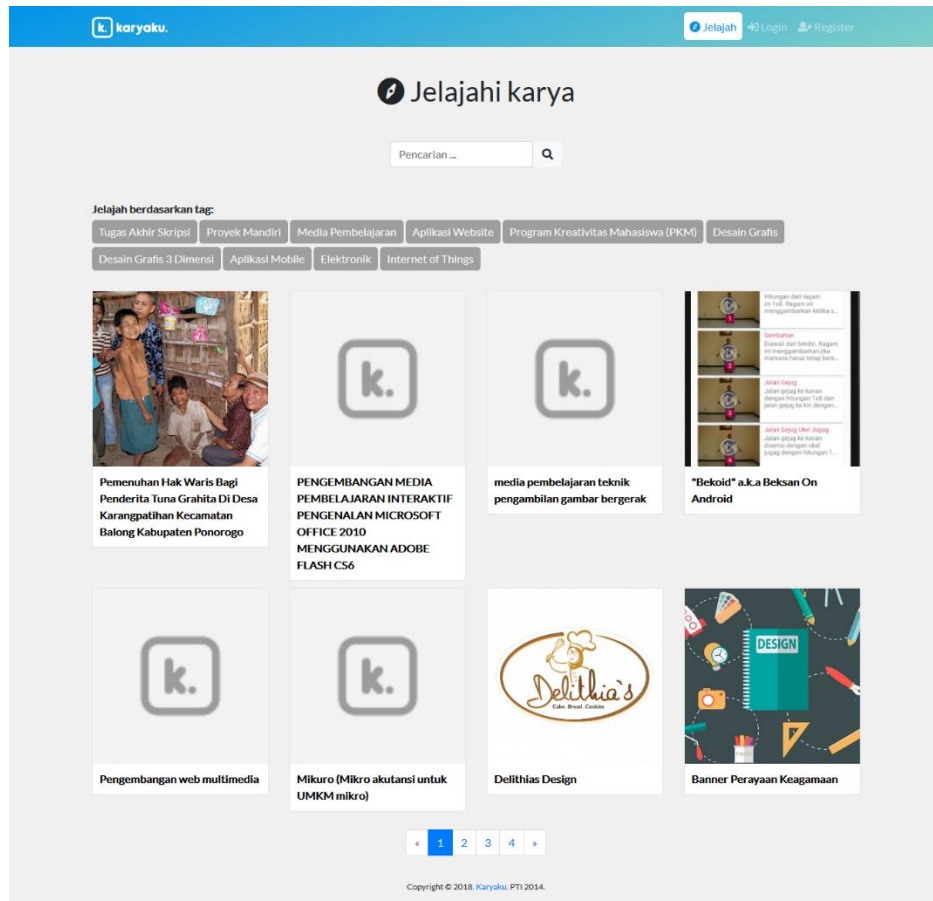


The screenshot shows the registration page of the Karyaku application. It features the same blue header and light gray background as the login page. The central white form is titled 'Pendaftaran Akun' (Account Registration). It contains several input fields: 'Nama Lengkap' (Full Name), 'Username', 'Alamat Email' (Email Address), 'Program Studi' (Study Program) which is a dropdown menu currently showing 'Manajemen Pendidikan - S1', 'Password', and 'Ulang Password' (Repeat Password). A blue 'Daftar' (Register) button is located at the bottom of the form. Below the form, there is a link that says 'Sudah memiliki akun? [Masuk](#)' (Already have an account? [Login](#)).

Gambar 24. Implementasi Halaman Register

Halaman *register* digunakan untuk mahasiswa mendaftarkan akun aplikasi. Halaman ini memuat beberapa masukan dari nama lengkap, *username*, *email*, program studi, dan *password*.

### (3) Halaman Jelajah Karya



Gambar 25. Implementasi Halaman Jelajah

Halaman jelajah karya memuat karya-karya terbaru yang diunggah oleh mahasiswa yang telah terdaftar dalam sistem. Selain itu, halaman jelajah karya terdapat *tag-tag* yang dapat menjelajah karya sesuai dengan *tag* yang telah ditambahkan oleh pengguna.

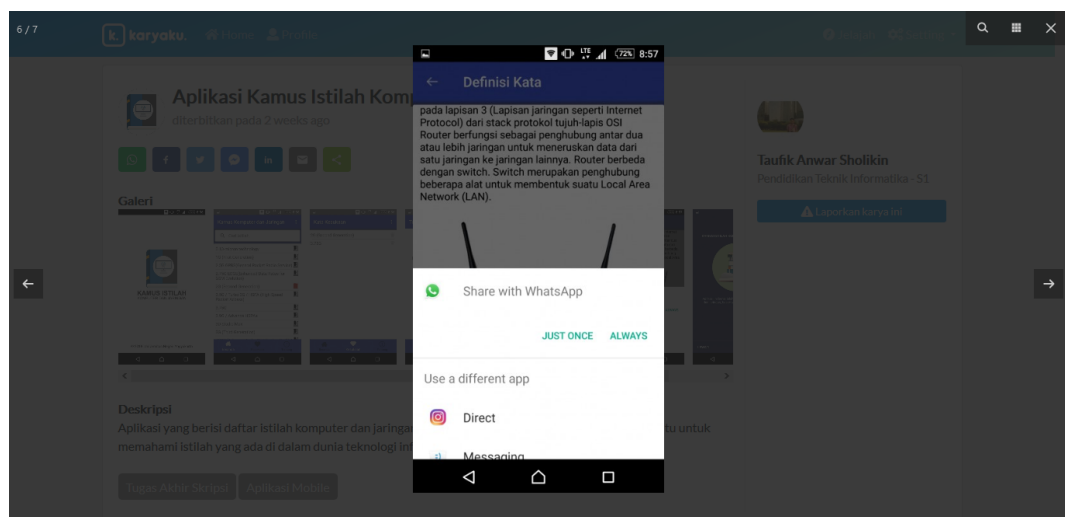


#### (4) Halaman Karya

Halaman karya memuat tampilan karya yang telah diunggah oleh pengguna mahasiswa. Informasi yang dimuat antara lain: judul karya, tombol berbagi ke media sosial, galeri karya (gambar dan video), deskripsi, *tag*, dan pengguna yang mengunggah. Halaman karya dapat dilihat pada Gambar 26. Galeri karya ditampilkan dengan *slideshow* seperti yang ditampilkan pada Gambar 27.



Gambar 26. Implementasi Halaman Karya

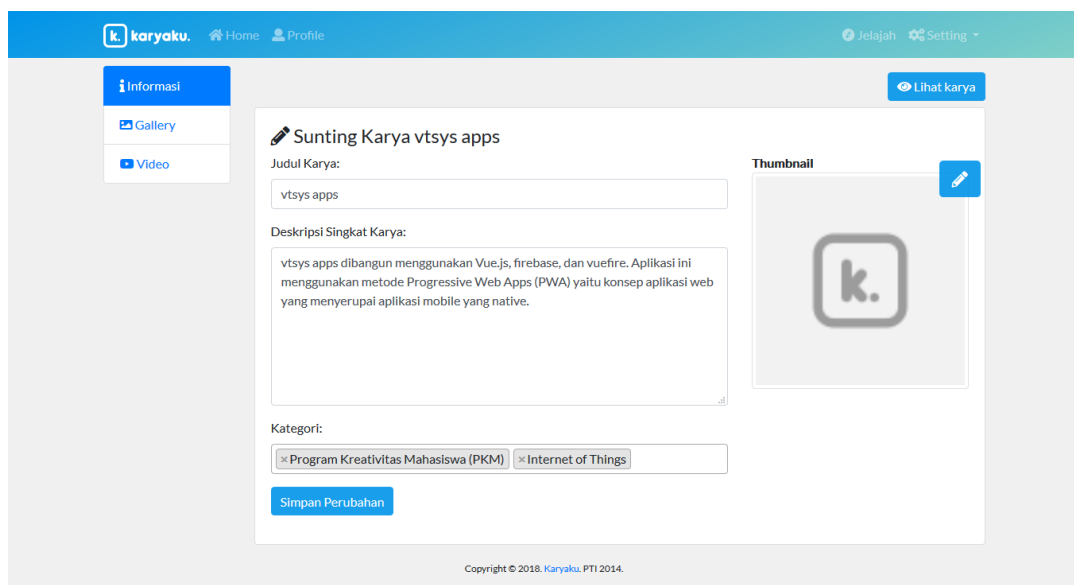


Gambar 27. Implementasi Halaman Karya dengan Galeri

#### (5) Halaman Sunting Karya

Halaman sunting karya merupakan halaman untuk mengelola karya yang telah diunggah. Halaman ini menyediakan 3 *tab* yang terdiri dari sunting informasi karya, galeri pendukung, dan video pendukung karya.

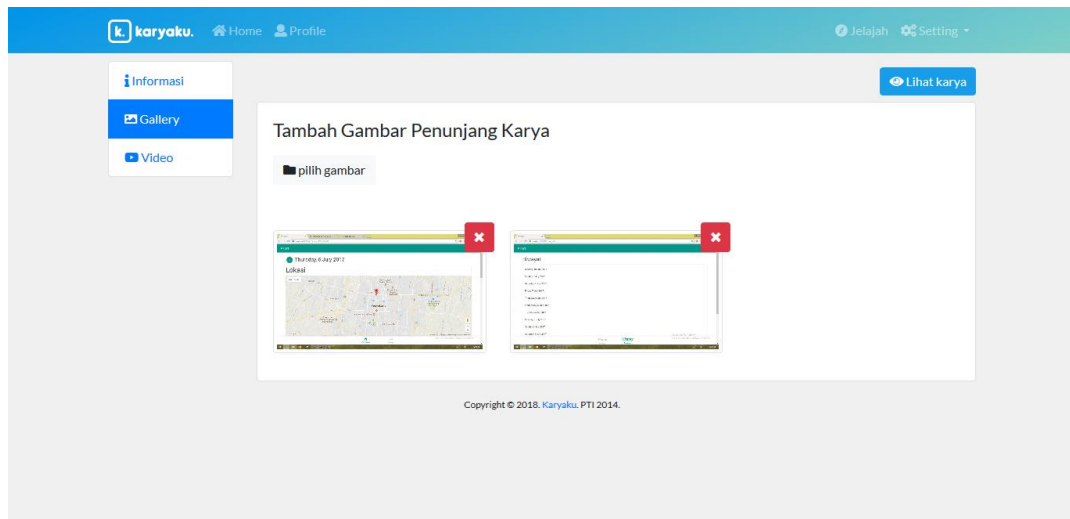
*Tab* yang pertama merupakan bagian untuk menyunting informasi dasar dari karya antara lain: judul karya, deskripsi karya, gambar *thumbnail* karya, dan penambahan tag karya. Bagian *tab* ini ditampilkan pada Gambar 28.



The screenshot displays the 'Sunting Karya' (Edit Work) interface. At the top, there's a navigation bar with 'karyaku.' logo, 'Home', 'Profile', 'Jelajah', and 'Setting'. A sidebar on the left contains 'Informasi' (selected), 'Gallery', and 'Video'. The main content area is titled 'Sunting Karya vtsys apps'. It features a 'Judul Karya:' field with the value 'vtsys apps', a 'Deskripsi Singkat Karya:' text area with the description 'vtsys apps dibangun menggunakan Vue.js, firebase, dan vuefire. Aplikasi ini menggunakan metode Progressive Web Apps (PWA) yaitu konsep aplikasi web yang menyerupai aplikasi mobile yang native.', and a 'Kategori:' section with two tags: 'Program Kreativitas Mahasiswa (PKM)' and 'Internet of Things'. A 'Thumbnail' section shows a placeholder image with a 'k.' logo and a 'Lihat karya' button. At the bottom, there's a 'Simpan Perubahan' button and a copyright notice: 'Copyright © 2018. Karyaku. PTI 2014.'

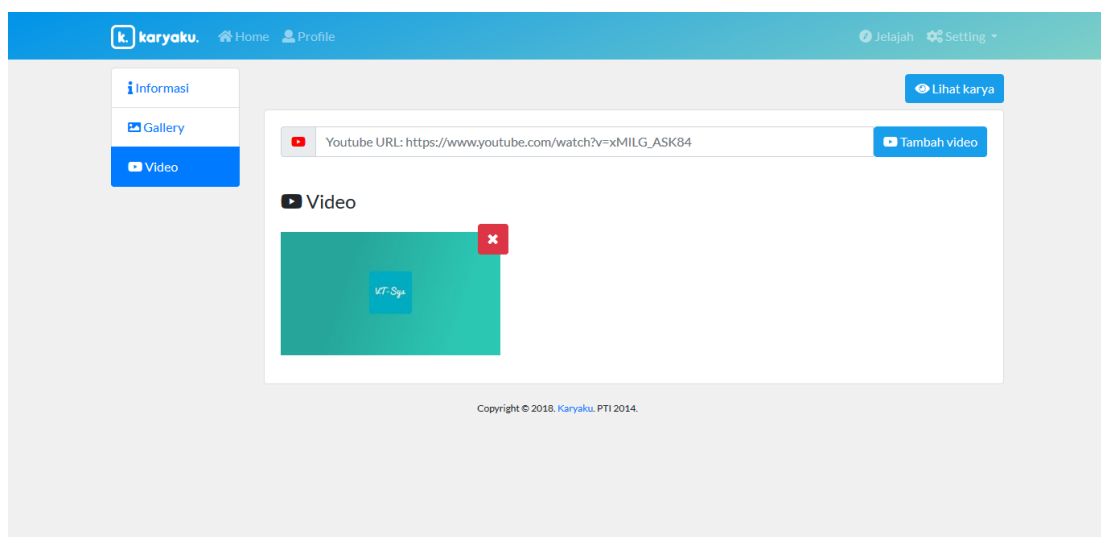
Gambar 28. Implementasi Halaman Sunting Karya

*Tab* yang kedua merupakan bagian untuk mengelola galeri karya yang berupa gambar. Pada *tab* ini, memiliki 2 bagian utama yaitu bagian *form* untuk mengunggah gambar dan bagian kelola gambar yang diunggah. Pada bagian bawah, terdapat gambar-gambar yang telah diunggah dan dapat dihapus oleh pengguna. Tampilan *tab* ini dapat dilihat pada Gambar 29.



Gambar 29. Implementasi Halaman Sunting Karya bagian Gallery

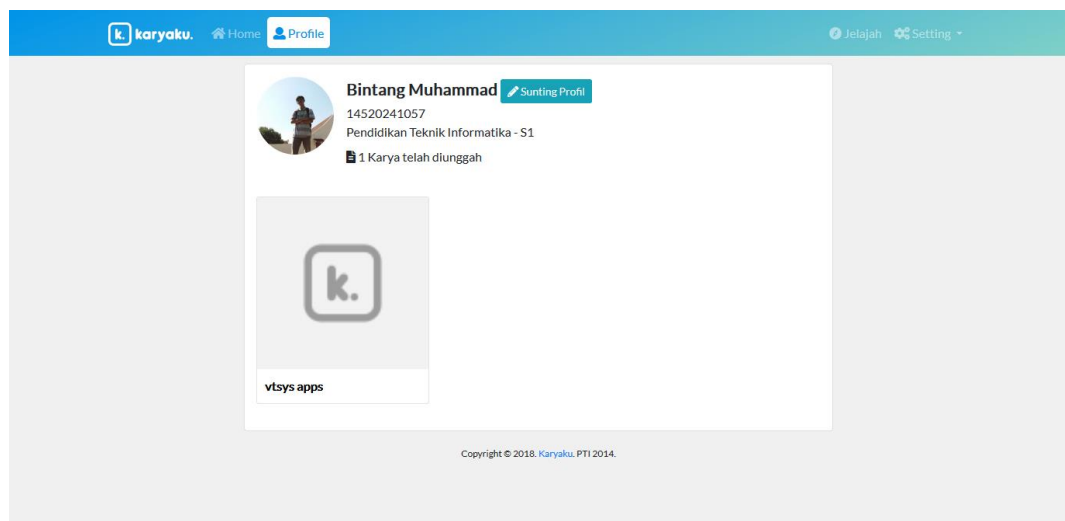
*Tab* yang ketiga merupakan bagian untuk mengelola video Youtube yang ditambahkan oleh pengguna untuk mendukung karya. Bagian halaman ini menyediakan *form* untuk memasukan tautan video Youtube dan bagian kelola video yang telah ditambahkan.. Pengguna juga dapat menghapus video yang telah ditambahkan di bagian kelola video ini dengan memilih tombol hapus dengan video yang sesuai. Tampilan bagian halaman ini dapat dilihat pada Gambar 30.



Gambar 30. Implementasi Halaman Karya bagian Video

#### (6) Halaman Profil

Halaman profil merupakan halaman yang menampilkan profil mahasiswa dan menampilkan karya-karya yang telah diunggah oleh mahasiswa tersebut. Halaman ini dapat diakses oleh pengunjung ataupun mahasiswa pengguna aplikasi ini. Halaman ini terdiri dari informasi mahasiswa, foto profil, dan bagian karya yang telah diunggah. Tampilan halaman profil dapat dilihat pada Gambar 31.

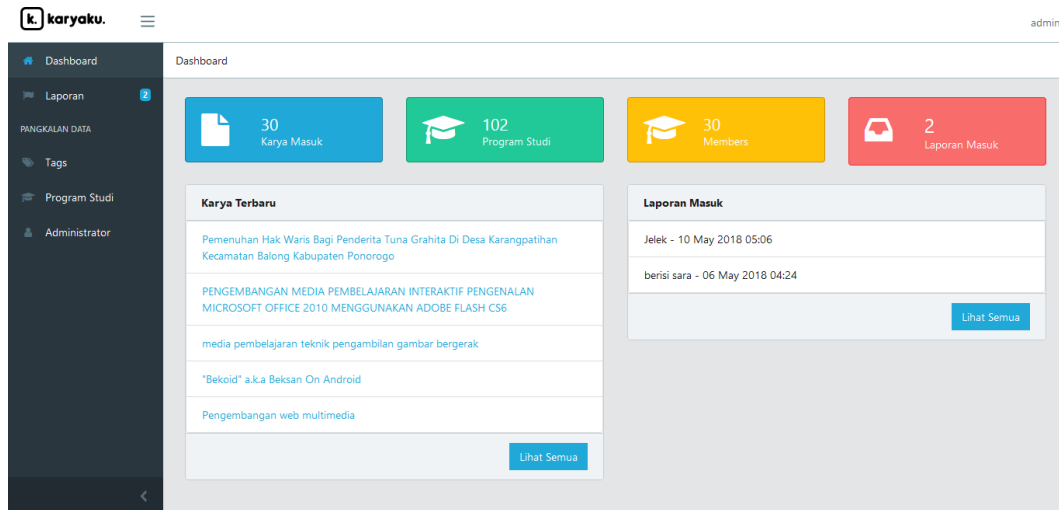


Gambar 31. Implementasi Halaman Profil

#### (7) Halaman *Dashboard* Admin

Halaman administrator merupakan halaman yang khusus diakses oleh pemegang akun administrator. Halaman administrator digunakan untuk mengelola data yang dibutuhkan aplikasi dari kelola data laporan, data program studi, data *tag* untuk karya, dan kelola akun administrator itu sendiri.

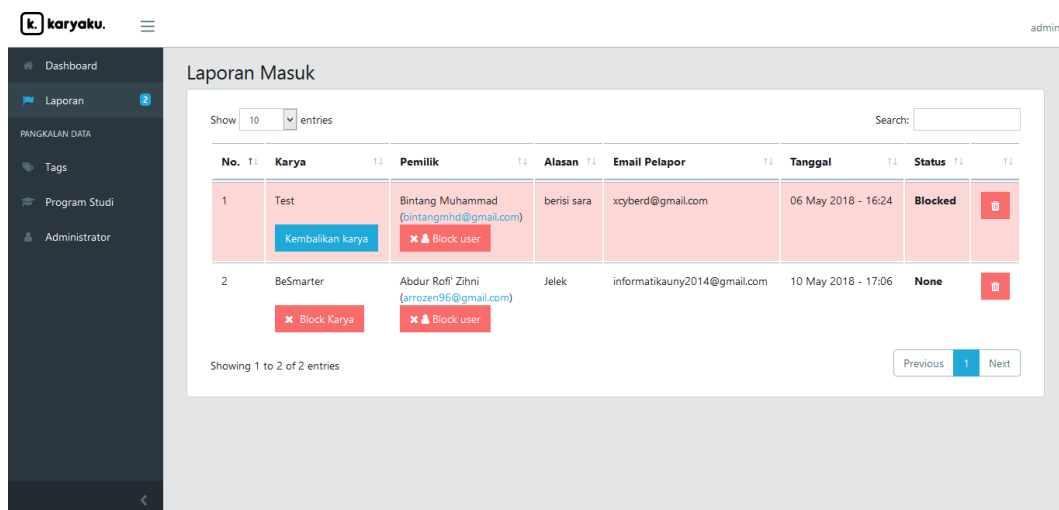
Halaman *dashboard* administrator merupakan halaman depan dari seluruh halaman administrator. Halaman ini berisi *counter* atau penghitung karya, program studi, mahasiswa yang telah mendaftar dan laporan yang telah masuk. Selain itu terdapat bagian karya yang baru masuk dan bagian laporan masuk yang ditampilkan secara sederhana. Tampilan dashboard dapat dilihat pada Gambar 32.



Gambar 32. Implementasi Halaman Dashboard Admin

#### (8) Halaman Laporan Admin

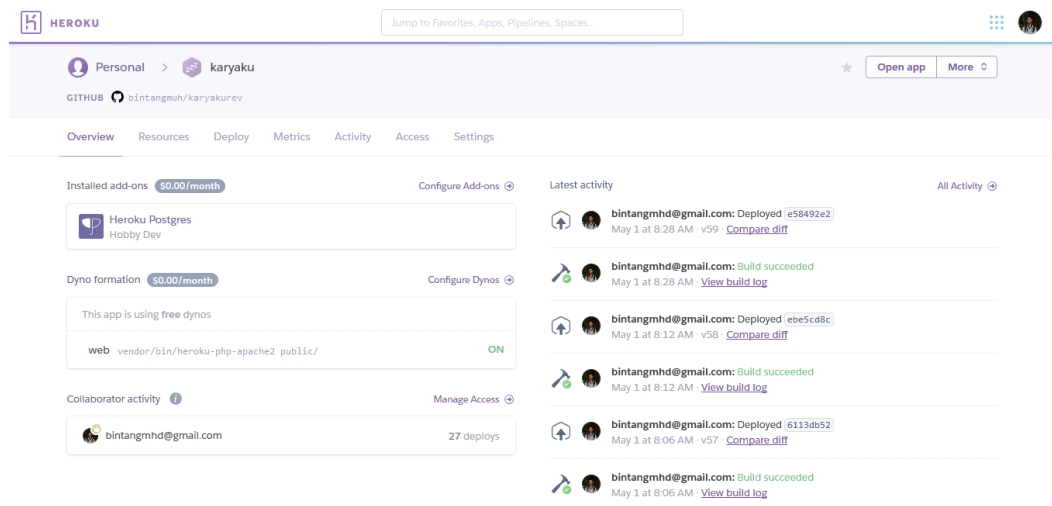
Halaman laporan admin merupakan salah satu halaman administrator yang mengelola laporan yang telah masuk dari pengguna. Pengguna yang memasukan laporan berisi email yang dapat dikontak dan alasan pelaporan karya. Tiap laporan terdapat tombol yang digunakan memblokir karya maupun pengguna. Admin juga dapat mengembalikan karya atau pengguna yang telah diblokir. Tampilan halaman laporan tesaji pada Gambar 33.



Gambar 33. Implementasi Halaman Laporan Admin

## 2) Deployment

Deployment pada tahap *construction* yaitu memasang aplikasi pada *cloud server* uji coba Heroku untuk dilaksanakan ujicoba aplikasi secara daring atau *online*. Aktivitas yang dilaksanakan cukup mengkonfigurasi aplikasi yang akan digunakan. Konfigurasi *cloud* Heroku ditunjukkan pada Gambar 34.



Gambar 34. Konfigurasi Heroku

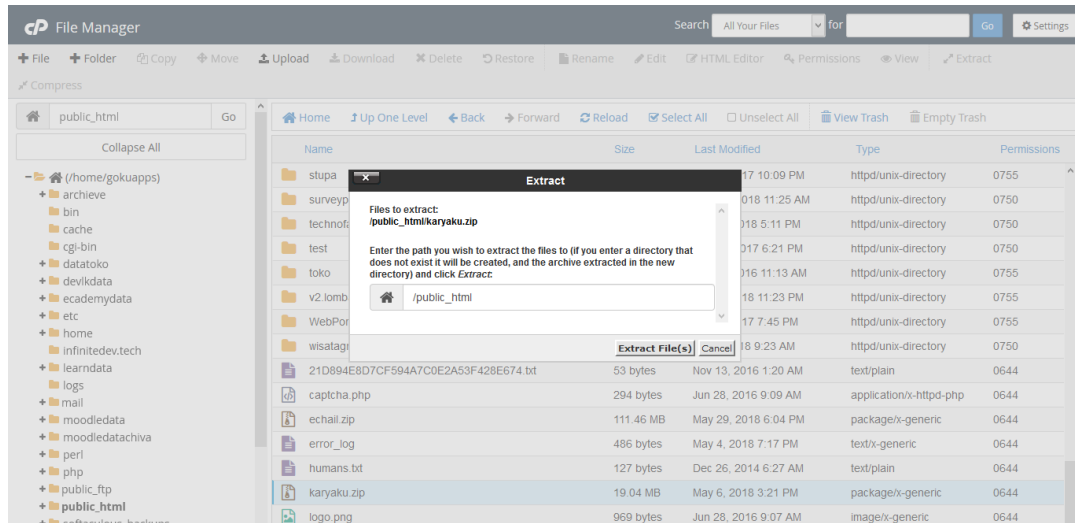
### d. Tahap Transition

Tahap *transition* merupakan tahap yang berfokus pada *deployment* yaitu pemasangan aplikasi pada server utama untuk dibuka secara umum dan *testing* yaitu menguji kelayakan dari aplikasi dengan menggunakan standar yang telah ditentukan. Tahap ini merupakan tahap akhir dari metode RUP.

## 1) Deployment

Kegiatan *deployment* pada tahap *transition* berfokus pada pemasangan pada *server* untuk produksi. Server yang digunakan untuk produksi berupa *shared hosting*. Konfigurasi yang dilakukan mulai dari pemasangan *source code* dan konfigurasi basis data yang akan digunakan. Tahap pengunggahan *source code*

pada *hosting* dapat dilihat pada Gambar 35. Hasil dari pemasangan ke *hosting* dapat dilihat di website Karyaku dengan tautan <http://karyaku.infiniteuny.id/>.



Gambar 35. Proses pengunggahan dan ekstraksi *source code* pada hosting

## 2) Test

Kegiatan *testing* pada tahap *transition* fokus kepada pengujian dengan melakukan *alpha testing* dan *beta testing* pada *production server*. *Testing* yang dilakukan supaya ketika digunakan pengguna sistem dalam keadaan siap.

## 2. Analisis Kualitas Aplikasi Web

Aplikasi web dianalisis kualitasnya menggunakan standar ISO 25010 dengan meliputi 5 aspek yaitu *functional suitability*, *usability*, *performance efficiency*, *reliability*, dan *maintainability*.

### a. *Functional Suitability*

Pengujian *functional suitability* menguji fungsional aplikasi yang telah direncanakan dapat berjalan dengan benar atau tidak. Pengujian *functional suitability* dilakukan oleh 2 orang yang ahli dalam pengembangan perangkat lunak

khususnya aplikasi website dan sistem informasi. Daftar ahli yang melakukan pengujian *functional suitability* dapat dilihat di Tabel 17.

Tabel 17. Penguji *Functional Suitability*

No.	Nama	Profesi	Instansi
1	Hilarius Widya Iswara, S.Pd.	<i>IT Developer</i>	Wongselo.net
2	Deni Kurnianto Nugroho, S.Pd.	<i>Fullstack Developer</i>	PT. Sisfomedika

Pengujian dilakukan dengan memberikan 24 butir fungsi yang perlu diuji. Tiap fungsi yang berjalan dengan sesuai menurut penguji maka diberikan nilai 1 dan yang tidak berjalan dengan sesuai diberikan nilai 2. Hasil pengujian yang dilakukan adalah sebagai berikut.

Tabel 18. Hasil Pengujian *Functional Suitability*

Nomor Pertanyaan	Berhasil	
	Ya	Tidak
1	2	0
2	2	0
3	2	0
4	2	0
5	2	0
6	2	0
7	2	0
8	2	0
9	2	0
10	2	0
11	2	0
12	2	0
13	2	0
14	2	0
15	2	0



Nomor Pertanyaan	Berhasil	
	Ya	Tidak
16	2	0
17	2	0
18	2	0
19	2	0
20	2	0
21	2	0
22	2	0
23	2	0
24	2	0
<b>Total</b>	48	0

$$X = 1 - \frac{A}{B}$$

$$X = 1 - \frac{0}{48}$$

Sehingga,

$$X = 1$$

**b. Usability**

Pengujian *usability* merupakan pengujian aspek kemudahan penggunaan aplikasi oleh pengguna. Pengujian *usability* ini menggunakan *USE Questionnaire* yang berjumlah 30 butir pernyataan dengan menggunakan skala likert dengan 5 rentang nilai. Pengujian ini melibatkan 28 responden mahasiswa Universitas Negeri Yogyakarta. Hasil pengujian *usability* terdapat pada Gambar 36.

No	Nama	Program Studi	Pernyataan																														Total	
			Usefulness								Ease of Use											Ease of Learning				Satisfaction								
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30		
1	Andri Muhyidin	PTI	5	4	4	3	3	4	4	4	5	5	4	4	3	4	4	4	4	4	4	4	4	5	5	4	4	4	4	4	5	123		
2	Herjuna Artanto	PTE	4	4	5	4	4	4	5	4	4	4	4	4	4	4	4	4	4	4	4	5	4	4	4	5	5	4	4	5	5	127		
3	Setyorini	PTI	5	5	5	4	4	4	4	4	5	4	4	5	5	5	4	4	4	4	5	5	5	5	5	5	5	4	4	4	4	134		
4	M. Arif Sulistya	PTI	4	4	4	5	4	5	4	4	4	4	5	4	4	5	4	4	5	4	4	5	4	4	5	5	4	5	5	5	5	133		
5	Septiani Audina	PTI	5	5	5	4	5	4	5	5	5	5	4	5	5	4	5	5	4	5	5	4	5	5	4	4	5	5	5	5	5	141		
6	Ratna Mutiara	PTI	4	4	4	4	4	5	4	4	5	5	4	4	5	4	5	4	4	4	4	4	5	4	5	4	4	4	5	4	5	130		
7	Gagas Akar Ilalang	PTI	4	4	4	2	2	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	2	2	4	4	3	4	2	109	
8	Anggie Ayu Wardani	PTI	4	4	5	5	5	4	5	4	5	4	5	5	4	4	4	5	4	5	5	5	5	5	4	4	4	4	5	4	4	5	135	
9	Intan Lisnawati	Matematika	4	3	5	3	4	2	4	3	3	3	3	4	3	3	3	4	4	3	4	3	4	4	3	4	4	4	3	3	3	4	104	
10	Galih Malela D	PTI	5	4	5	5	4	5	4	5	3	5	4	5	4	5	5	4	4	4	5	5	4	5	4	5	4	5	4	5	4	5	135	
11	Taufik Anwar Solikin	PTI	4	4	5	4	5	5	5	4	5	5	5	5	5	5	5	5	4	5	5	5	5	5	5	5	5	5	4	5	5	4	143	
12	Mumtaz Fuadi Ahmad	PTI	5	5	5	4	4	5	5	5	5	5	5	5	4	5	5	5	5	5	5	5	5	5	5	5	4	5	4	4	4	5	143	
13	Ilham Bagoes T	PTI	4	4	5	4	4	5	3	4	5	4	4	4	3	4	4	4	4	4	5	5	4	4	4	4	4	4	5	5	4	4	125	
14	Anjasmoro Adi N	PTI	5	4	5	4	4	5	4	4	4	4	4	5	4	4	4	4	4	4	4	4	4	4	4	4	4	3	4	4	4	5	124	
15	Linda Noviasari	PTI	5	5	4	4	5	4	4	5	4	4	4	5	5	4	4	4	5	4	5	5	4	5	5	4	5	4	4	4	4	5	133	
16	Ahmad Aisar Fahmi	PTI	4	4	5	4	5	5	4	3	5	5	5	5	5	5	5	4	4	4	5	5	5	5	5	4	5	4	4	5	4	5	137	
17	Fuad Reza Pahlevi	PTI	4	4	5	5	4	4	4	4	5	4	4	4	4	4	4	4	4	5	5	5	5	4	4	4	4	4	4	4	4	4	126	
18	Isnainul Fahrizal	PTI	3	4	5	4	4	4	4	4	5	5	5	5	4	5	5	5	4	5	5	5	5	5	4	4	4	4	4	4	4	4	132	
19	Singgih Bekt	PT Mekatronika	4	4	4	4	3	3	4	4	5	4	4	4	4	4	4	4	4	3	4	5	5	5	5	4	4	4	3	5	5	4	4	123
20	Permadi Afrian	PTI	4	4	5	4	4	3	4	4	5	5	5	5	4	5	5	4	3	5	5	5	5	5	5	5	4	4	4	4	5	4	133	
21	Hernawan Prabowo	PTE	4	4	5	4	4	4	5	4	4	4	4	5	4	4	4	5	4	4	5	5	5	5	5	5	4	4	5	5	4	5	133	
22	Ilham Anasrulloh	PTI	4	2	5	4	3	4	4	3	3	4	4	4	3	3	4	3	3	3	4	5	5	4	4	4	4	3	5	4	5	114		
23	Jumiyati	PTI	5	5	5	5	5	5	5	5	5	5	4	5	5	5	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	148	
24	Lutfi Rohmatulloh	PTI	4	3	5	3	4	5	4	4	4	4	4	4	4	4	5	4	4	4	4	4	4	4	4	4	5	4	4	4	5	4	123	
25	Tiara Putra	PTI	4	4	4	3	5	4	4	4	4	4	4	4	3	4	4	4	4	4	4	4	4	4	4	4	5	4	4	4	5	4	121	
26	Aditya Nur Riskan Nugroho	PTI	3	4	4	2	4	5	4	4	5	5	4	5	4	5	5	5	4	5	5	5	5	5	5	4	4	4	5	5	4	4	132	
27	Ni Putu Nuratni	PTI	4	4	5	4	5	3	3	4	4	3	4	2	4	4	5	5	3	4	4	5	4	4	4	4	4	4	3	3	4	4	117	
28	Ersyana Candra Dewi	PPKn	5	3	4	2	2	2	3	4	5	5	5	5	4	3	3	4	4	5	4	5	4	5	5	5	3	4	3	5	5	4	120	
Total			119	112	131	107	113	116	116	114	125	122	119	125	115	118	120	121	113	118	127	132	126	128	124	120	118	115	118	122	120	124	3598	
Nilai Maksimal			140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	4200	

Gambar 36. Hasil Pengujian *Usability*

Berdasarkan tabel hasil *USE Questionnaire* diatas dapat diketahui bahwa jawab **sangat setuju (SS)** berjumlah **319**, **setuju (S)** berjumlah **452**, **ragu-ragu (RG)** berjumlah **57**, **tidak setuju (TS)** berjumlah **12**, dan **sangat tidak setuju (STS)** berjumlah **0**. Data yang telah dikumpulkan dimasukkan ke dalam rumus:

$$\text{Skor}_{\text{TOTAL}} = (319 \times 5) + (452 \times 4) + (57 \times 3) + (12 \times 2) + (0 \times 1)$$

$$\text{Skor}_{\text{TOTAL}} = 3598$$

Lalu dihitung persentase *usability* dengan menggunakan rumus sebagai berikut:

$$\text{Persentase} = \frac{3598}{30 \times 28 \times 5} \times 100$$

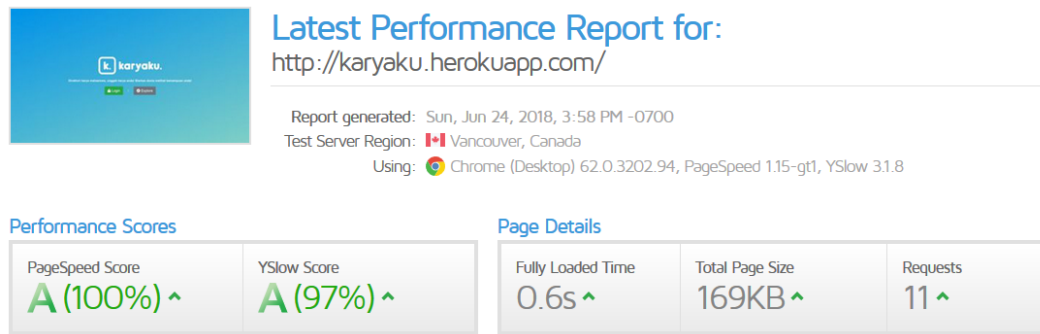
$$\text{Persentase} = \frac{3598}{4200} \times 100$$

$$\text{Persentase} = 85.67 \%$$

Hasil persentase pengujian *usability* adalah **85.67%**. Menurut klasifikasi interpretasi nilai pada Tabel 6, dapat dinyatakan bahwa hasil pengujian ini menunjukan aplikasi **sangat layak** dan memenuhi aspek *usability*.

### c. **Performance Efficiency**

Pengujian *performance efficiency* mengukur keefisienan waktu aplikasi dalam memuat halaman website. Pengujian aspek ini menggunakan *tools* GTMetrix. Hasil pengujian halaman depan tersaji pada Gambar 37.



Gambar 37. Hasil Pengujian *Performance Efficiency* Halaman Depan

Hasil dari pengujian keseluruhan aplikasi terlampir pada **Lampiran 12** dan hasilnya dapat dilihat pada Tabel 19.

Tabel 19. Tabel Hasil Pengujian *Performance Efficiency*

No.	Tampilan Aplikasi	Pagespeed (%)	Yslow (%)	Waktu (detik)
1	Halaman Depan	100	97	0,6
2	Halaman Login	98	96	0.5
3	Halaman Register	98	94	0,7
4	Halaman Beranda	97	90	0,9
5	Halaman Unggah Karya	98	95	0,8
6	Halaman Pencarian	100	94	0,6
7	Halaman Jelajah	100	94	1,1
8	Halaman Karya	96	85	1,4
9	Halaman Sunting Karya	99	89	0,8
10	Halaman Pelaporan Karya	100	94	1
11	Halaman Sunting Profil	100	93	1
12	Halaman Profil	100	94	0,7
13	Halaman Login Admin	97	93	0,8
14	Halaman <i>Dashboard</i>	96	89	0,9
15	Halaman Kelola Laporan	96	87	1,4
16	Halaman Kelola <i>Tags</i>	96	87	0,9
17	Halaman Kelola Prodi	96	87	1,4

No.	Tampilan Aplikasi	Pagespeed (%)	Yslow (%)	Waktu (detik)
18	Halaman Kelola Administrator	96	89	1
<b>Rerata</b>		97.94	91.50	0,91

Dari hasil pengujian *performance efficiency*, dapat dilihat bahwa rata-rata waktu pemuatan halaman website adalah **0,91** detik. Penilaian Google Pagespeed dengan rata-rata bernilai **97,94%** yang berarti memiliki **Grade A** sedangkan penilaian Yahoo Yslow sebesar **91.50%** yang setara **Grade A**.

#### d. *Reliability*

Pengujian *reliability* aplikasi web Karyaku dilakukan dengan menggunakan perangkat lunak WAPT yang berfungsi untuk melakukan *stress testing* pada aplikasi berbasis website. Hasil pengujian *reliability* tersaji pada Gambar 38.

##### Test execution parameters:

Test status: finished  
 Test started at: 6/4/2018 1:40:14 PM  
 Scenario name: test2.wps  
 Test run comment:  
 Test executed by: bintang (HIDRO-PC)  
 Test executed on: localhost  
 Test duration: 0:10:00

##### Summary

Profile	Successful sessions	Failed sessions	Successful pages	Failed pages	Successful hits	Failed hits
Profile2	2894	0	2894	0	23122	57

Gambar 38. Hasil Pengujian *Reliability* Menggunakan WAPT

Hasil pengujian *reliability* dapat dilihat pada *summary* gambar diatas berupa perhitungan *sessions*, *pages*, dan *hits*. Successfull sessions sebesar 2894 dan failed session sebesar 0. *Succesfull pages* sebesar 2894 dan *failed pages* sebesar 0. Sedangkan untuk *hits* mendapatkan 23122 untuk *successful hits* dan 57 untuk *failed hits*.

#### e. *Maintainability*

Pengujian *maintainability* aplikasi web Karyaku dilakukan dengan menggunakan *tools* PHPMetrics yang berfungsi untuk menghitung *maintainability index*. Hasil pengujian *maintainability* tersaji pada Gambar 39.

Name	Lines of code	Logical lines of code	Comment weight	Maintainability index	Bugs	Lack of cohesion of methods
 (9)	6994	686	17.04	125.71	0.07	1.13

Gambar 39. Hasil Pengujian *Maintainability* Menggunakan PHPMetrics

Dari hasil pengujian *maintainability* aplikasi web Karyaku mendapatkan *Maintainability Index* sebesar **125.71**. PHPMetrics memberikan evaluasi nilai *maintainability* 100 dari 100 seperti yang ditunjukkan pada Gambar 40.

Factor	Score
Maintainability	100 / 100
Accessibility for new developers	50 / 100
Simplicity of algorithms	96.71 / 100
Volume	100 / 100
Reducing bug's probability	100 / 100

Gambar 40. Hasil Pengujian *Maintainability* dengan PHPMetrics

## B. Pembahasan

Bagian ini akan membahas hasil pengujian aplikasi web Karyaku sebagai direktori karya mahasiswa menggunakan standar ISO 25010 yang meliputi 5 aspek dari *functional suitability*, *usability*, *performance efficiency*, *reliability*, dan *maintainability*.

### **1. Aspek *Functional Suitability***

Hasil pengujian aspek *functional suitability* pada aplikasi web Karyaku sebagai direktori karya mahasiswa menunjukkan nilai  $X=1$ . Nilai  $X$  menandakan bahwa fungsi sistem yang berjalan semakin baik jika mendekati nilai 1. Aplikasi ini memiliki nilai 1 sehingga fungsi dapat berjalan sesuai secara keseluruhan. Pernyataan di atas menyimpulkan bahwa aplikasi web Karyaku direktori karya mahasiswa telah memenuhi aspek *functional suitability*.

### **2. Aspek *Usability***

Hasil pengujian aspek *usability* pada aplikasi web Karyaku sebagai direktori karya mahasiswa menunjukkan persentase *usability* sebesar 85.67% atau memiliki tingkat *usability* sangat baik. Hasil pengujian ini membuktikan bahwa aplikasi web Karyaku telah memenuhi aspek *usability*.

### **3. Aspek *Performance Efficiency***

Hasil pengujian aspek *performance efficiency* pada aplikasi web Karyaku sebagai direktori karya mahasiswa menunjukkan rata-rata waktu pemuatan halaman website sebesar 0,91 detik. Web diklasifikasikan cepat oleh Nielsen jika waktu pemuatan halaman kurang dari 10 detik. Selain itu, aplikasi web Karyaku memiliki rata-rata nilai Pagespeed sebesar 97.94% yang setara Grade A dan rata-rata nilai Yslow 91,5% sebesar setara Grade A. Aplikasi web Karyaku dapat disimpulkan telah memenuhi aspek *performance efficiency*.

### **4. Aspek *Reliability***

Hasil pengujian aspek *reliability* pada aplikasi web Karyaku sebagai direktori karya mahasiswa menggunakan perangkat lunak WAPT menunjukkan hasil pada Tabel 20.

Tabel 20. Hasil Pengujian *Reliability* menggunakan WAPT

<b>Metrik</b>	<b>Berhasil</b>	<b>Gagal</b>	<b>Persentase Keberhasilan</b>
Session	2894	0	100%
Pages	2894	0	100%
Hits	23122	57	99,75%
<b>Rata-rata</b>			99,91%

Berdasarkan hasil yang telah didapatkan diatas aplikasi web Karyaku sebagai direktori karya mahasiswa memiliki persentase 100% untuk *session*, 100% untuk *page*, dan 99,75% untuk *hits*. Rata-rata keberhasilan dalam pengujian *reliability* sebesar 99,91% Untuk memenuhi standar Telcordia, aplikasi ini harus memiliki persentasi keberhasilan minimal 95%. Oleh karena itu, aplikasi web Karyaku sebagai direktori karya mahasiswa telah memenuhi aspek *reliability*.

## **5. Aspek *Maintainability***

Hasil pengujian aspek *maintainability* pada aplikasi web Karyaku sebagai direktori karya mahasiswa menggunakan *tools* PHPMetrics menunjukan nilai *Maintainability Index* (MI) sebesar 125,71. Dalam pengklasifikasian tingkat *maintainability index* pada Tabel 7, nilai 125,71 termasuk *High Maintainable* sehingga aplikasi ini mudah untuk dirawat. Oleh karena itu, aplikasi web Karyaku sebagai direktori karya mahasiswa telah memenuhi aspek *maintainability*.



## **BAB V**

### **Kesimpulan dan Saran**

#### **A. Kesimpulan**

Berdasarkan hasil penelitian dan pembahasan yang telah dilaksanakan, dapat disimpulkan bahwa:

1. Aplikasi web Karyaku telah memenuhi kebutuhan untuk menjadi media informasi publikasi hasil karya mahasiswa secara daring ditunjukan dengan kesesuaian antara kebutuhan pengguna (*requirement*) dengan fitur yang dikembangkan dan fitur telah diuji kelengkapannya pada aspek pengujian *functional suitability*. Pengembangan aplikasi web Karyaku menggunakan model pengembangan *Rational Unified Process* (RUP) yang terdiri dari *inception, elaboration, construction, dan transistion*. Aplikasi ini dikembangkan menggunakan *framework* Laravel.
2. Aplikasi web Karyaku sebagai direktori karya mahasiswa telah teruji menggunakan Standar ISO 25010 yang terdiri dari 5 aspek. Hasil pengujian aspek *functional suitability* mendapatkan nilai 1 atau fungsi berjalan benar secara keseluruhan, aspek *usability* dinyatakan sangat layak mendapatkan nilai 85,67%, aspek *performance efficiency* terpenuhi dengan waktu pemuatan 0,91 detik, aspek *reliability* dinyatakan memenuhi syarat dengan rata-rata keberhasilan 99,91%, dan aspek *maintainability* dinyatakan mudah dirawat dengan maintainabiliy index 125,71.

## **B. Keterbatasan Produk**

Produk yang dikembangkan dalam penelitian ini memiliki keterbatasan produk sebagai berikut:

1. Kurangnya fitur publikasi karya yang dilakukan secara kelompok.
2. Belum tersedia fitur apresiasi (*like*, suka, atau *love*) atau komentar dari sesama pengguna mahasiswa.
3. Belum tersedia fitur umpan balik atau komentar dari sesama pengguna mahasiswa.
4. Belum tersedianya fitur pengelompokan karya per program studi.

## **C. Pengembangan Produk Lebih Lanjut**

Pengembangan produk lebih lanjut dapat dilakukan dengan cara menambahkan fitur baru yang membuat aplikasi lebih interaktif seperti fitur umpan balik (komentar) dan fitur apresiasi (*like*, suka, atau *love*), fitur untuk publikasi karya dilakukan secara berkelompok, serta fitur pengelompokan karya per program studi.

## **D. Saran**

Berdasarkan kesimpulan penelitian yang telah dilaksanakan dan keterbatasan produk yang telah dikembangkan peneliti memberi saran sebagai berikut:

1. Aplikasi sebaiknya ditambahkan fitur apresiasi dan umpan balik supaya aplikasi menjadi lebih interaktif antar pengguna aplikasi bisa saling berkomentar mengenai karya-karya yang diunggah pengguna lain.
2. Aplikasi sebaiknya memiliki fitur pengunggahan karya yang telah dikerjakan berkelompok.

## DAFTAR PUSTAKA

- Ajzele, B. (2016 ). *Modular Programming with PHP 7*. Brimingham: Packt Publishing.
- Asiatun, K. (2011). Pengembangan Database Evaluasi Diri Jurusan Pendidikan Teknik Boga dan Busana Ft Uny. *Jurnal Pendidikan Teknologi dan Kejuruan Vol. 20 No. 2*.
- Asthana, A., & Olivieri, J. (2009). Quantifying software reliability and readiness. In Communications Quality and Reliability. *IEEE International Workshop Technical Committee*, 1-6.
- Bala, A., & Chhillar, R. (2016). Automatic Test Data Generation using Genetic Algorithm using Sequence Diagram. *International Journal of Computer Systems*.
- Dyer, R. J. (2015). *Learning MySQL and MariaDB*. California: O'Reilly Media, Inc.
- Erwin, H., & Salam, A. (2014). PERANCANGAN WEBSITE SEBAGAI GALERI PROGRAM STUDI TEKNIK INFORMATIKA D3 MULTIMEDIA UNIVERSITAS DIAN NUSWANTORO. *Skripsi, Fakultas Ilmu Komputer*.
- Fowler, M. (2004). *UML Distilled Edisi 3*. Yogyakarta: Andi.
- Georgiana, V. (2011). Aplikasi Virtual Gallery Berbasiskan Content Management System pada Jurusan Sistem Informasi Universitas Bina Nusantara. *Comtech Volume 2*.
- Hariyanto, D. (2008). Pengembangan Sistem Informasi Akademik Mahasiswa Berbasis Teknologi WAP (Wireless Application Protocol) di Jurusan Pendidikan Teknik Elektro FT UNY. *Jurnal Pendidikan dan Teknologi Kejuruan Vol. 17, No.2*.
- Irmawati, D., & Indrihapsari, Y. (2014). Sistem Informasi Kearsipan untuk Meningkatkan Kualitas Pelayanan. *Jurnal Pendidikan Teknologi dan Kejuruan Vol. 22 No. 2*, 136-147.
- Johnson, R. S., Mims-Cox, J., & Doyle-Nichols, A. (2010). *Developing Portfolios in Education Second Edition*. California: Sage Publications.
- Kappel, G., Proll, B., & Reich, S. (2003). *Web Engineering (The Discipline of Systematic Development of Web Applications)*. Heidelberg: John Wiley & Sons Ltd.

- Kaur, S., Kaur, K., & Kaur, P. (2016). An Empirical Performance Evaluation of Universities Website. *International Journal of Computer Applications Volume 146 No.15, July 2016*.
- Krasner, G. E., & Stephen, T. (1988). A Description of the Model-View-Controller User Interface Paradigm in the Smalltalk-80 System. *Journal of Object Oriented Programming*, 26-49.
- Kromann, F. M. (2016). *PHP and MySQL Recipes: A Problem-Solution Approach Second Edition*. Springer: California.
- Kruchten, P. (2004). *The Rational Unified Process, Third Edition*. Massachusetts: Addison-Wesley Professional.
- Kusumah, W., & Dwigatama, D. (2009). *Mengenal Penelitian Tindakan Kelas*. Jakarta: Indeks.
- Laravel. (2018, January 28). *Blade Templates*. Diambil kembali dari Laravel: <https://laravel.com/docs/5.6/blade>
- Laravel. (2018, January 28). *Controllers*. Diambil kembali dari Laravel: <https://laravel.com/docs/5.6/controllers>
- Laravel. (2018, January 28). *Eloquent: Getting Started*. Diambil kembali dari Laravel: <https://laravel.com/docs/5.6/eloquent>
- Logofătu, M. G. (2006). e-Portfolio in student's evaluation. *Proceedings of The 1st International Conference on Virtual Learning (ICVL)*.
- Lowe, D., & Pressman, R. (2009). *Web Engineering: A Practitioner's Approach*. New York: McGraw-Hill.
- Lund, A. (2001). Measuring usability with the use questionnaire. *Usability interface*.
- Najm, N. M. (2014). Measuring Maintainability Index of a Software Depending on Line of Code Only. *IOSR Journal of Computer Engineering (IOSR-JCE) vol. 16, Issue 2*, 64-69.
- Nielsen, J. (2010, June 21). *Website Response Times*. Diambil kembali dari Nielsen Norman Group: <https://www.nngroup.com/articles/website-response-times/>

- Nielsen, J. (2012, Juni 4). *How Many Test Users in a Usability Study?* Dipetik April 4, 2018, dari Nielsen Norman Group:  
<https://www.nngroup.com/articles/how-many-test-users/>
- Perlman, G. (2015). *USE Questionnaire: Usefulness, Satisfaction, and Ease of use*. Diambil kembali dari <http://garyperlman.com/>:  
<http://garyperlman.com/quest/quest.cgi?form=USE>
- Pitt, C. (2012). *Pro PHP MVC*. UK: Apress.
- Pradhan, D. (2013, February 7). *WAPT; A Load Testing Tool That Delivers! [Review]*. Dipetik February 25, 2017, dari Software Testing Tricks:  
<http://www.softwaretestingtricks.com/2013/02/best-load-stress-performance-testing-tool-WAPT.html>
- Pressman, R. (2014). *Software Engineering: A Prationer's Approach 8th edition*. New York: Mc-Graw and Hill.
- Rosa A.S., & Shalahuddin, M. (2011). *Rekayasa Perangkat Lunak*. Bandung: Modula.
- Rumbaugh, J., Jacobson, I., & Booch, G. (2004). *Unified modeling language reference manual*. Boston: Pearson Higher Education.
- Setiawan, D. (2011). Perancangan dan Pembuaan Pangkalan Data Portofolio Mahasiswa. *Seminar Nasional Teknologi Informasi*.
- Sommerville, I. (2011). *Software Engineering Ninth edition*. Boston: Addison-Wesley.
- Stauffer, M. (2016). *Laravel Up and Running*. California: O'Reilly Media inc.
- Sudaryono, Guritno, S., & Rahardja, U. (2011). *Theory and Application of IT Research (Metodologi Penelitian Teknologi Informasi)*. Yogyakarta: Penerbit Andi.
- Sugiyono. (2006). *Metode Penelitian Pendidikan*. Bandung: Alfabeta.
- Surguy, M. (2014). *Laravel: My First Framework*. Brimingham: Leanpub.
- Suryn, W. (2014). *Software Quality Engineering: A Practitioner's Approach*. New Jersey: John Wiley & Sons, Inc.

W3Techs. (2018, February 23). *Usage of server-side programming languages for websites*. Diambil kembali dari W3Techs - Web Technology Survey:  
[https://w3techs.com/technologies/overview/programming\\_language/all](https://w3techs.com/technologies/overview/programming_language/all)

Wagner, S. (2013). *Software Product Quality Control*. Springer: New York.

Wazlawick, R. S. (2014). *Object-Oriented Analysis and Design for Information Systems*. Massachusetts: Elsevier Inc.

# LAMPIRAN

## **Daftar Lampiran**

- Lampiran 1. Surat Keputusan Pembimbing Skripsi
- Lampiran 2. Lembar Persetujuan Proposal Skripsi
- Lampiran 3. Surat Pernyataan Validasi Instrumen
- Lampiran 4. Hasil Angket Survey Kepemilikan Karya Mahasiswa
- Lampiran 5. Skenario Use Case
- Lampiran 6. Activity Diagram
- Lampiran 7. Sequence Diagram
- Lampiran 8. Desain (wireframe) Antarmuka Aplikasi.
- Lampiran 9. Implementasi Antarmuka Aplikasi
- Lampiran 10. Hasil Pengujian Functional Suitability
- Lampiran 11. Hasil Pengujian Usability
- Lampiran 12. Hasil Pengujian Performance Efficiency
- Lampiran 13. Foto Kegiatan



## Lampiran 1. Surat Keputusan Pembimbing Skripsi

**KEPUTUSAN DEKAN FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA  
NOMOR : 69/PINF/PB/IV/2018**

**TENTANG  
PENGANGKATAN DOSEN PEMBIMBING TUGAS AKHIR SKRIPSI (TAS) MAHASISWA  
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

**DEKAN FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

- Menimbang : a. bahwa untuk kelancaran pelaksanaan kegiatan Tugas Akhir Skripsi (TAS) mahasiswa, dipandang perlu mengangkat dosen pembimbingnya;  
b. bahwa untuk keperluan sebagaimana dimaksud pada huruf a perlu menetapkan Keputusan Dekan Tentang Pengangkatan Dosen Pembimbing Tugas Akhir Skripsi (TAS) Fakultas Teknik Universitas Negeri Yogyakarta.
- Mengingat : 1. Undang-undang RI Nomor 20 Tahun 2003 Tentang Sistem Pendidikan Nasional (Lembaran Negara Tahun 2003 Nomor 78, Tambahan Lembaran Negara Nomor 4301);  
2. Peraturan Pemerintah Republik Indonesia Nomor 4 Tahun 2014 Tentang Penyelenggaraan Pendidikan Tinggi dan Pengelolaan Perguruan Tinggi (Lembaran Negara Tahun 2014 Nomor 16, Tambahan Lembaran Negara Republik Indonesia Nomor 5500);  
3. Keputusan Presiden Republik Indonesia Nomor 93 Tahun 1999 Tentang Perubahan Institut Keguruan dan Ilmu Pendidikan menjadi Universitas;  
4. Peraturan Mendiknas RI Nomor 23 Tahun 2011 Tentang Organisasi dan Tata Kerja Universitas Negeri Yogyakarta;  
5. Peraturan Mendiknas RI Nomor 34 Tahun 2011 Tentang Statuta Universitas Negeri Yogyakarta;  
6. Keputusan Menteri Pendidikan dan Kebudayaan RI Nomor 98/MPK.A4/KP/2013 Tentang Pengangkatan Rektor Universitas Negeri Yogyakarta;  
7. Peraturan Rektor Nomor 2 Tahun 2014 tentang Peraturan Akademik;  
8. Keputusan Rektor Nomor 800/UN.34/KP/2016 tahun 2016 tentang Pengangkatan Dekan Fakultas Teknik Universitas Negeri Yogyakarta.

**MEMUTUSKAN**

Menetapkan : **KEPUTUSAN DEKAN TENTANG PENGANGKATAN DOSEN PEMBIMBING TUGAS AKHIR SKRIPSI (TAS) FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA.**

**PERTAMA** : Mengangkat Saudara :

Nama	: Nurkhamid, S.Si., M.Kom., Ph.D.
NIP	: 19680707 199702 1 001
Pangkat/Golongan	: Penata Muda, III/a
Jabatan Akademik	: Asisten Ahli

sebagai Dosen Pembimbing Untuk mahasiswa penyusun Tugas Akhir Skripsi (TAS) :

Nama	: Bintang Muhammad
NIM	: 14520241057
Prodi Studi	: Pend. Teknik Informatika - S1
Judul Skripsi/TA	: PENGEMBANGAN APLIKASI WEB KARYAKU SEBAGAI DIREKTORI KARYA MAHASISWA UNIVERSITAS NEGERI YOGYAKARTA

- KEDUA : Dosen Pembimbing sebagaimana dimaksud dalam Diktum PERTAMA bertugas merencanakan, mempersiapkan, melaksanakan, dan mempertanggungjawabkan pelaksanaan kegiatan bimbingan terhadap mahasiswa sebagaimana dimaksud dalam Diktum PERTAMA sampai mahasiswa dimaksud dinyatakan lulus.
- KETIGA : Biaya yang diperlukan dengan adanya Keputusan ini dibebankan pada Anggaran DIPA Universitas Negeri Yogyakarta Tahun 2018.
- KEEMPAT : Keputusan ini berlaku sejak tanggal 2 April 2018.

Tembusan Keputusan Dekan ini disampaikan kepada :

1. Para Wakil Dekan Fakultas Teknik;
  2. Kepala Bagian Tata Usaha Fakultas Teknik;
  3. Kepala Subbagian Keuangan dan Akuntansi Fakultas Teknik;
  4. Kepala Subbagian Pendidikan Fakultas Teknik;
  5. Mahasiswa yang bersangkutan;
- Universitas Negeri Yogyakarta.

Ditetapkan di : Yogyakarta  
Pada tanggal : 2 April 2018

DEKAN FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA,



Dr. Drs. WIDARTO, M.Pd.  
NIP. 19631230 198812 1 001

## Lampiran 2. Lembar Persetujuan Proposal Skripsi

**HALAMAN PERSETUJUAN**

Proposal penelitian yang berjudul "Pengembangan Aplikasi Web Karyaku Sebagai Direktori Karya Mahasiswa Universitas Negeri Yogyakarta" yang disusun oleh Bintang Muhammad, NIM: 14520241057 telah disetujui oleh pembimbing.

Yogyakarta, April 2018

<p>Ketua Jurusan Pendidikan Teknik Elektronika &amp; Informatika</p>  <p><u>Dr. Fatchul Azzafin, S.T., M.T.</u> NIP. 19720508 199802 1 002</p>	<p>Dosen Pembimbing</p>  <p><u>Nurkhamid, S.Si., M.Kom., Ph.D</u> NIP. 19680707 199702 1 001</p>
---	--

Mengetahui,

Fakultas Teknik UNY



Dr. Wigarto, M.Pd.  
NIP. 19631230 198812 1 001



### Lampiran 3. Surat Pernyataan Validasi Instrumen

#### SURAT PERNYATAAN VALIDASI INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan di bawah ini :

Nama : Handaru Jati, ST., M.M, M.T, Ph.D.  
NIP : 19740511 199903 1 002  
Jurusan : Pendidikan Teknik Elektronika dan Informatika

Menyatakan

Nama : Bintang Muhammad  
NIM : 14520241057  
Program Studi : Pendidikan Teknik Informatika  
Judul TAS : Pengembangan Aplikasi Web Karyaku Sebagai Direktori Karya Mahasiswa Universitas Negeri Yogyakarta

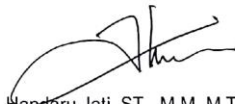
Setelah dilakukan kajian atas instrumen penelitian TAS tersebut dapat dinyatakan :

- ☒ Layak digunakan untuk penelitian  
☐ Layak digunakan dengan perbaikan  
☐ Tidak layak digunakan untuk penelitian yang bersangkutan

Dengan saran/perbaikan sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana mestinya.

Yogyakarta, 29 April 2018  
Validator,

  
Handaru Jati, ST., M.M, M.T, Ph.D.  
NIP. 19740511 199903 1 002

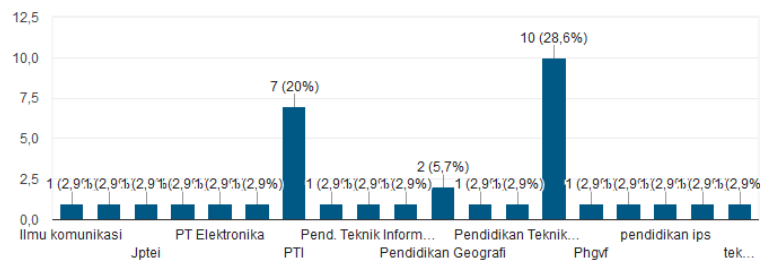
Catatan :

☐ beri tanda ✓

## Lampiran 4. Hasil Angket Survey Kepemilikan Karya Mahasiswa

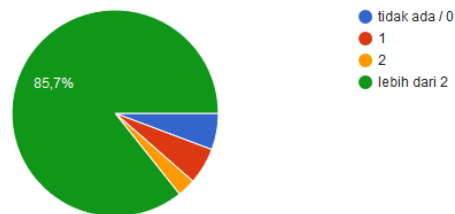
### Program Studi

35 tanggapan



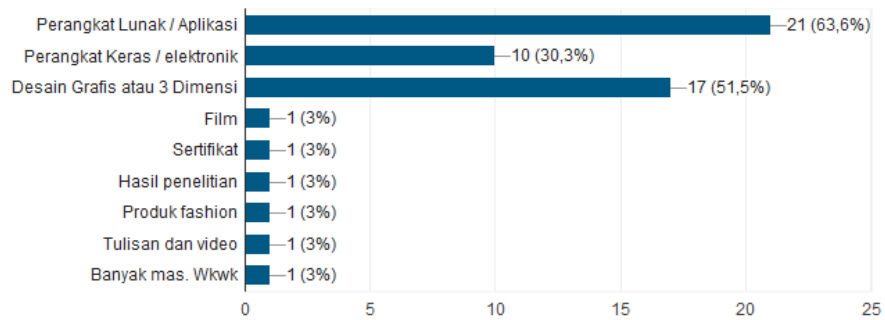
### Selama anda kuliah, berapa jumlah karya/proyek yang anda buat? (tugas kuliah, PKM, riset pribadi, dll.)

35 tanggapan



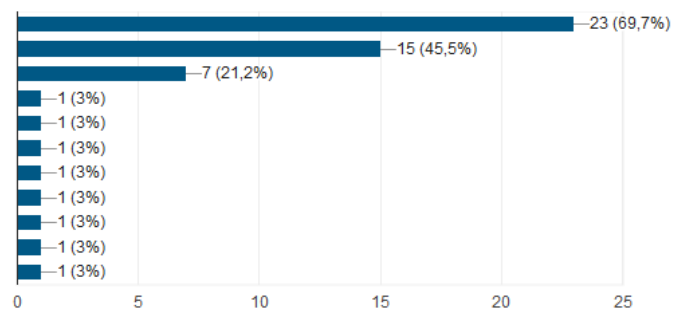
### Apa saja bentuk karya yang anda telah buat?

33 tanggapan



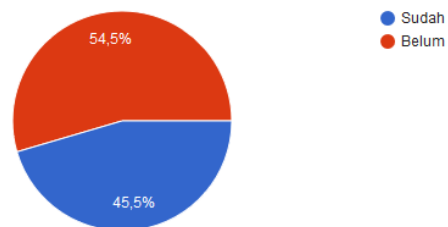
### Bagaimana anda mendokumentasikan dan mempublikasikan karya anda?

33 tanggapan



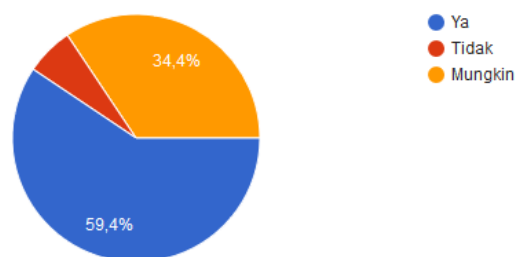
Apakah anda sudah mendokumentasikan dan mempublikasikan karya anda secara daring/online?

33 tanggapan



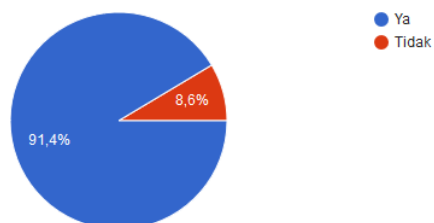
Apakah anda merasa kesulitan ketika mencari referensi karya dari mahasiswa di UNY?

32 tanggapan



Apakah kampus perlu memiliki sistem publikasi karya seperti masterpiece.its.ac.id, behance.net, dan sebagainya?

35 tanggapan



## Apa saran fitur pengembangan aplikasi web direktori karya mahasiswa

35 tanggapan

Mudah diakses (2)
Agar bisa digunakan selama perkuliahan
Mantapp, lanjutkannnn
Bisa mencantumkan gambar dan video
Memfasilitasi mhs untuk dokumentasi karya agar dapat dijadikan referensi dalam membuat inovasi lain yang lebih bermanfaat
Apa yah
Fitur sugestion karya, fitur seperti apa yang kita suka maka muncul beberapa karya yang sesuai dengan keinginan atau refrensi kita seperti Pinterest
Ada fitur pencarian karya-karya mahasiswa yang sudah ada Ada fitur upload karya sendiri Ada fitur share karya sendiri Ada fitur legalitas atau pembuatan HAKI untuk karya
Pencarian karya berdasar kategori, gambaran atau video tentang karya yg dibuat, share karya
Fitur untuk membuat CV berdasar karya yg di input
Penggunaannya yang lebih sederhana dan terintegrasi dengan berbagai website resmi UNY
-
nama karya, deskripsi, pemilik, tombol2 apresiasi dan komentar.
Bisa memberi masukan untuk karya yang ada dan bisa secara langsung melihat karya secara detail
Penjelasan aplikasi yang jelas, disertakan video atau screenshot dari tiap karya
Karya yang diupload bisa berbagai karya, tidak hanya satu bidang. Jadi karya yang dipublikasi bisa KTI, Essay, Artikel dan yang lainnya.
Di tambahan fitur2 yg membangkitkan & menumbuh kembangkan minat & inspirasi mahasiswa dalam berkarya (masukkan quotes2 yg menggugah semangat berkarya juga pencapaian & manfaat apa yg di dapatkan setelah membuat karya)
fitur diskusi
selain menyediakan list karya mahasiswa, beri juga cara publikasi yang baik serta untuk mencari hak paten agar karya tidak di duplikasi (plagiat) oleh pihak yang tidak bertanggung jawab
Detail karya yang dibuat
-
Semoga bisa membuat yang simple dan mudah digunakan siapa saja
kek webnya.....😭 aku lupa lamannya apa namanya 😊
Yang penting bisa untuk sharing karya, bahkan mungkin bisa untuk "pameran karya online" pada waktu-waktu khusus
bisa di jadiin portofolio
UI/UX yang menarik, diberikan leveling / experience
Kalau bisa menggunakan sistem rating, dimana mahasiswa dapat melihat karya karya yang dinilai layak
Hmmm apa ya mas 😊 dibanyakin fitur berguna yang mudah dipakai 😊 ya gitu
ada fitur chatnya untuk tanya tanya sama si pemilik karya
Hmm, apa yaa. Ada fitur klasifikasi karya yang jelas
Belum ada saran..
Kategorisasi berdasarkan jenis karya



Lampiran 5. Skenario *Use Case*

1. Skenario *Use Case* Mahasiswa dan Pengunjung

a. Login

Tabel 21. Skenario *Use Case* Login

<b>Nama Use Case</b>	Login
<b>Aktor</b>	Pengunjung
<b>Deskripsi</b>	<i>Use case</i> untuk masuk kedalam aplikasi sebagai mahasiswa
<b>Pre Condition</b>	-
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Pengguna membuka halaman login.</li> <li>2. Sistem menampilkan <i>form</i> login menggunakan email dan password.</li> <li>3. Pengguna memasukan email dan password lalu mengeklik submit.</li> <li>4. Sistem memvalidasi kredensial login.</li> <li>5. Sistem menampilkan halaman beranda.</li> </ol>
<b>Exception</b>	<ol style="list-style-type: none"> <li>4a. Sistem tidak menemukan kredensial login dalam basis data.</li> <li>4a. 1. Sistem mengembalikan aktor ke halaman login</li> <li>4a. 2. Sistem menampilkan pesan kesalahan</li> <li>4a. 3. Pengguna kembali ke proses 3</li> </ol>
<b>Post Condition</b>	Aktor pengguna menjadi aktor mahasiswa

b. Regsiter

Tabel 22. Skenario *Use Case* Register

<b>Nama Use Case</b>	Regsiter
<b>Aktor</b>	Pengunjung
<b>Deskripsi</b>	<i>Use case</i> untuk mendaftar akun sebagai mahasiswa
<b>Pre Condition</b>	Pengunjung masuk kedalam sistem
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Pengguna membuka halaman register.</li> </ol>

	2. Sistem menampilkan <i>form</i> registrasi. 3. Pengguna memasukkan data-data sesuai <i>form</i> . 4. Pengguna mengeklik submit. 5. Sistem memvalidasi masukan dari pengguna. 6. Sistem menyimpan ke basis data
<b>Exception</b>	5a. Sistem memberikan kegagalan validasi. 5a. 1. Sistem mengembalikan aktor ke halaman login 5a. 2. Sistem menampilkan pesan kesalahan 5a. 3. Pengguna kembali ke proses 3
<b>Post Condition</b>	Aktor pengguna menjadi aktor mahasiswa

c. Jelajah Karya

Tabel 23. Skenario *Use Case* Jelajah Karya

<b>Nama Use Case</b>	Jelajah Karya
<b>Aktor</b>	Pengunjung dan Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan penjelajahan karya secara keseluruhan atau berdasarkan <i>tag</i>
<b>Pre Condition</b>	-
<b>Main Flow</b>	1. Pengguna memasuki halaman jelajah 2. Sistem menampilkan karya-karya terbaru dan tag 3. Pengguna memilih tag. 4. Sistem menampilkan karya sesuai tag
<b>Exception</b>	-
<b>Post Condition</b>	Pengguna dapat melihat karya-karya

d. Pencarian Karya

Tabel 24. Skenario *Use Case* Pencarian Karya

<b>Nama Use Case</b>	Pencarian Karya
<b>Aktor</b>	Pengunjung atau Mahasiswa

<b>Deskripsi</b>	<i>Use case</i> untuk melakukan pencarian karya berdasarkan kata kunci yang telah dimasukkan oleh pengguna.
<b>Pre Condition</b>	-
<b>Main Flow</b>	1. Pengguna memasuki halaman jelajah 2. Pengguna memasukan karta kunci pencarian 3. Sistem menampilkan karya-karya sesuai dengan kata kunci
<b>Exception</b>	3a. Sistem tidak menemukan karya sesuai dengan kata kunci 3a. 1. Sistem menampilkan pesan bahwa karya tidak ditemukan sesuai dengan kata kunci 3a. 2. Pengguna kembali ke proses 2.
<b>Post Condition</b>	Pengguna dapat melihat karya-karya sesuai dengan kata kunci

e. Lihat Karya

Tabel 25. Skenario *Use Case* Lihat Karya

<b>Nama Use Case</b>	Lihat Karya
<b>Aktor</b>	Pengunjung atau Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk melihat karya
<b>Pre Condition</b>	-
<b>Main Flow</b>	1. Pengguna memilih karya. 2. Sistem menampilkan informasi karya dari deskripsi, gambar, dan video
<b>Exception</b>	2a. Sistem tidak menemukan karya yang dipilih pengguna. 2a. 1. Sistem menampilkan halaman tidak ditemukan. 2a. 2. Pengguna dapat kembali ke halaman depan.
<b>Post Condition</b>	Pengguna dapat melihat karya

f. Bagikan Karya

Tabel 26. Skenario *Use Case* Bagikan Karya

<b>Nama Use Case</b>	Bagikan Karya
<b>Aktor</b>	Pengunjung atau Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk membagikan karya oleh pengguna
<b>Pre Condition</b>	-
<b>Main Flow</b>	1. Pengguna memilih karya. 2. Sistem menampilkan informasi karya dari deskripsi, gambar, dan video 3. Pengguna memilih sosial media yang diinginkan. 4. Sistem memunculkan <i>popup</i> sesuai dengan sosial media yang dipilih.
<b>Exception</b>	2a. Sistem tidak menemukan karya yang dipilih pengguna. 2a. 1. Sistem menampilkan halaman tidak ditemukan. 2a. 2. Pengguna dapat kembali ke halaman depan.
<b>Post Condition</b>	Pengguna dapat melihat <i>popup</i> sosial media yang sesuai

g. Laporkan Karya

Tabel 27. Skenario *Use Case* Laporkan Karya

<b>Nama Use Case</b>	Laporkan Karya
<b>Aktor</b>	Pengunjung atau Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk melaporkan karya oleh pengguna
<b>Pre Condition</b>	-
<b>Main Flow</b>	1. Pengguna mengeklik tombol laporkan karya. 2. Sistem menampilkan <i>form</i> pelaporan karya sesuai dengan karya. 3. Pengguna memasukkan alasan pelaporan dan mengeklik laporkan. 4. Sistem memvalidasi input.

	5. Sistem menyimpan laporan dari pengguna.
<b>Exception</b>	2a. Sistem tidak menemukan karya yang dipilih pengguna. 2a. 1. Sistem menampilkan halaman tidak ditemukan. 2a. 2. Pengguna dapat kembali ke halaman depan.
	4a. Sistem gagal dalam validasi input. 4a. 1. Sistem mengembalikan ke <i>form</i> pelaporan dengan pesan kesalahan. 4a. 1. Pengguna kembali ke proses 2.
<b>Post Condition</b>	Laporan masuk ke dalam administrator

#### h. Sunting Informasi Karya

Tabel 28. Skenario *Use Case* Sunting Informasi Karya

<b>Nama Use Case</b>	Sunting Informasi Karya
<b>Aktor</b>	Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk menyunting karya oleh mahasiswa
<b>Pre Condition</b>	1. Aktor mahasiswa terautentikasi. 2. Karya diunggah aktor mahasiswa.
<b>Main Flow</b>	1. Mahasiswa masuk kedalam halaman sunting karya. 2. Sistem menampilkan <i>form</i> informasi karya yang telah dibuat. 3. Mahasiswa menyunting informasi pada <i>form</i> yang diperlukan. 4. Pengguna mengeklik tombol simpan. 6. Sistem memvalidasi input pengguna. 5. Sistem menyimpan perubahan dalam basis data.
<b>Exception</b>	6a. Sistem gagal dalam validasi input. 6a. 1. Sistem mengembalikan ke <i>form</i> informasi karya dengan pesan kesalahan. 6a. 1. Pengguna kembali ke proses 3.
<b>Post Condition</b>	Informasi dasar karya sudah tersunting

i. Unggah Gambar

Tabel 29. Skenario *Use Case* Unggah Gambar

<b>Nama Use Case</b>	Unggah Gambar
<b>Aktor</b>	Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk mengunggah gambar mendukung galeri karya oleh mahasiswa
<b>Pre Condition</b>	1. Aktor mahasiswa terautentikasi. 2. Karya diunggah aktor mahasiswa.
<b>Main Flow</b>	1. Mahasiswa masuk kedalam halaman sunting karya bagian galeri. 2. Sistem menampilkan <i>form</i> file gambar untuk karya yang telah dibuat. 3. Mahasiswa memilih <i>file</i> gambar. 4. Pengguna melakukan perintah unggah. 5. Sistem memvalidasi dan mengolah input <i>file</i> gambar dari pengguna. 6. Sistem menyimpan gambar yang diunggah kedalam <i>file system</i> . 7. Sistem menyimpan nama file ke dalam basis data. 8. Sistem menampilkan gambar yang telah diunggah.
<b>Exception</b>	5a. Sistem gagal memvalidasi dan mengolah input <i>file</i> gambar dari pengguna. 5a. 1. Sistem menampilkan pesan kesalahan. 5a. 2. Pengguna kembali ke proses 3.
<b>Post Condition</b>	Gambar galeri telah terunggah

j. Hapus Gambar

Tabel 30. Skenario *Use Case* Hapus Gambar

<b>Nama Use Case</b>	Hapus Gambar
----------------------	--------------

<b>Aktor</b>	Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk menghapus gambar pendukung karya oleh mahasiswa
<b>Pre Condition</b>	1. Aktor mahasiswa terautentikasi. 2. Karya diunggah aktor mahasiswa.
<b>Main Flow</b>	1. Mahasiswa masuk kedalam halaman sunting karya bagian galeri. 2. Sistem menampilkan file gambar yang telah terunggah untuk karya yang telah dibuat. 3. Mahasiswa memilih <i>file</i> gambar dan melakukan perintah hapus. 4. Sistem menghapus gambar
<b>Exception</b>	-
<b>Post Condition</b>	Gambar galeri telah dihapus

k. Tambah Video

Tabel 31. Skenario *Use Case* Tambah Video

<b>Nama Use Case</b>	Tambah Video
<b>Aktor</b>	Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk menambahkan video pendukung karya oleh mahasiswa
<b>Pre Condition</b>	1. Aktor mahasiswa terautentikasi. 2. Karya diunggah aktor mahasiswa.
<b>Main Flow</b>	1. Mahasiswa masuk kedalam halaman sunting karya bagian video. 2. Sistem menampilkan form url untuk tautan Youtube. 3. Mahasiswa memasukkan tautan Youtube. 4. Sistem memvalidasi tautan Youtube. 5. Sistem menyimpan ke dalam basis data.
<b>Exception</b>	4a. Sistem gagal memvalidasi tautan Youtube yang tidak valid.

	4a. 1. Sistem menampilkan pesan kesalahan. 4a. 2. Mahasiswa kembali ke proses 3.
<b>Post Condition</b>	Video Youtube ditambahkan ke dalam karya

#### l. Hapus Video

Tabel 32. Skenario *Use Case* Hapus Video

<b>Nama Use Case</b>	Hapus Video
<b>Aktor</b>	Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk menghapus video pendukung karya oleh mahasiswa
<b>Pre Condition</b>	1. Aktor mahasiswa terautentikasi. 2. Karya diunggah aktor mahasiswa.
<b>Main Flow</b>	1. Mahasiswa masuk kedalam halaman sunting karya bagian video. 2. Sistem menampilkan video-video yang telah terunggah untuk karya yang telah dibuat. 3. Mahasiswa memilih video Youtube dan menekan tombol hapus. 4. Sistem menghapus video
<b>Exception</b>	-
<b>Post Condition</b>	Video Youtube dihapus dari dalam karya

#### m. Sunting Informasi profil

Tabel 33. Deskripsi *Use Case* Sunting Informasi Profil

<b>Nama Use Case</b>	Sunting Informasi Profil
<b>Aktor</b>	Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk mengelola informasi dasar profil mahasiswa
<b>Pre Condition</b>	Aktor mahasiswa terautentikasi.



<b>Main Flow</b>	1. Mahasiswa masuk kedalam halaman sunting profil. 2. Sistem menampilkan form sunting profil. 3. Mahasiswa memasukan data untuk edit profil dan menekan tombol simpan. 4. Sistem memvalidasi input pengguna. 5. Sistem menyimpan ke dalam basis data.
<b>Exception</b>	4a. Sistem gagal memvalidasi input. 4a. 1. Sistem menampilkan pesan kesalahan. 4a. 2. Mahasiswa kembali ke proses 3.
<b>Post Condition</b>	Data profil tersunting

n. Ganti gambar profil

Tabel 34. Skenario *Use Case* Ganti Gambar Profil

<b>Nama Use Case</b>	Ganti Gambar Profil
<b>Aktor</b>	Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk mengelola gambar profil mahasiswa
<b>Pre Condition</b>	Aktor mahasiswa terautentikasi.
<b>Main Flow</b>	1. Mahasiswa masuk kedalam halaman sunting profil. 2. Sistem menampilkan form unggah gambar pada sunting profil. 3. Mahasiswa memilih <i>file</i> gambar dan menekan tombol unggah. 4. Sistem memvalidasi input <i>file</i> gambar dari pengguna. 5. Sistem menyimpan gambar ke <i>file system</i> . 6. Sistem menyimpan nama <i>file</i> ke dalam basis data.
<b>Exception</b>	4a. Sistem gagal memvalidasi input <i>file</i> gambar. 4a. 1. Sistem menampilkan pesan kesalahan. 4a. 2. Mahasiswa kembali ke proses 3.
<b>Post Condition</b>	Gambar profil terganti

o. Ganti Password

Tabel 35. Skenario *Use Case* Ganti Password

<b>Nama Use Case</b>	Ganti Password
<b>Aktor</b>	Mahasiswa
<b>Deskripsi</b>	<i>Use case</i> untuk mengganti <i>password</i> akun profil mahasiswa
<b>Pre Condition</b>	Aktor mahasiswa terautentikasi.
<b>Main Flow</b>	1. Mahasiswa masuk kedalam halaman sunting profil. 2. Sistem menampilkan form penggantian password pada sunting profil. 3. Mahasiswa memasukan password baru sebanyak 2 kali. 4. Sistem memvalidasi password pengguna. 6. Sistem menyimpan password pengguna ke dalam basis data.
<b>Exception</b>	4a. Sistem gagal memvalidasi password baru pengguna. 4a. 1. Sistem menampilkan pesan kesalahan. 4a. 2. Mahasiswa kembali ke proses 3.
<b>Post Condition</b>	Password telah terganti

2. Deskripsi *Use Case* Administrator

a. Login Admin

Tabel 36. Skenario *Use Case* Login Admin

<b>Nama Use Case</b>	Login Admin
<b>Aktor</b>	Pengunjung
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan login ke dalam akun administrator aplikasi
<b>Pre Condition</b>	-
<b>Main Flow</b>	1. Pengguna membuka halaman login admin.

	2. Sistem menampilkan <i>form</i> login menggunakan username dan password. 3. Pengguna memasukan email dan password lalu mengeklik submit. 4. Sistem memvalidasi kredensial login. 5. Sistem menampilkan halaman beranda/dashboard admin.
<b>Exception</b>	4a. Sistem tidak menemukan kredensial login admin dalam basis data. 4a. 1. Sistem mengembalikan aktor ke halaman login 4a. 2. Sistem menampilkan pesan kesalahan 4a. 3. Pengguna kembali ke proses 3
<b>Post Condition</b>	Administrator terautentifikasi

b. Tambah Tag

Tabel 37. Skenario *Use Case* Tambah Tag

<b>Nama Use Case</b>	Tambah Tag
<b>Aktor</b>	Administrator
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan penambahan <i>tag</i> karya yang akan disediakan oleh aplikasi
<b>Pre Condition</b>	Administrator sudah terautentifikasi
<b>Main Flow</b>	1. Administrator memasukan data <i>tag</i> terbaru. 2. Adminsitrator melakukan perintah simpan. 3. Sistem memvalidasi input administrator. 4. Sistem menyimpan ke dalam basis data.
<b>Exception</b>	3a. Sistem gagal melakukan validasi. 3a. 1. Sistem menampilkan pesan kesalahan 3a. 2. Pengguna kembali ke proses 2
<b>Post Condition</b>	<i>Tag</i> baru ditambahkan

c. Hapus Tag

Tabel 38. Skenario *Use Case* Hapus Tag

<b>Nama Use Case</b>	Hapus Tag
<b>Aktor</b>	Administrator
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan penghapusan <i>tag</i> karya yang akan disediakan oleh aplikasi
<b>Pre Condition</b>	Administrator sudah terautentifikasi
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Administrator memilih data <i>tag</i>.</li> <li>2. Adminsitrator melakukan perintah hapus <i>tag</i>.</li> <li>3. Sistem memvalidasi tag yang akan dihapus.</li> <li>4. Sistem menghapus <i>tag</i> dari basis data.</li> </ol>
<b>Exception</b>	<ol style="list-style-type: none"> <li>3a. Sistem gagal melakukan validasi.</li> <li>3a. 1. Sistem menampilkan pesan kesalahan</li> <li>3a. 2. Pengguna kembali ke proses 2</li> </ol>
<b>Post Condition</b>	<i>Tag</i> telah terhapus

d. Tambah Admin

Tabel 39. Skenario *Use Case* Tambah Admin

<b>Nama Use Case</b>	Tambah Admin
<b>Aktor</b>	Administrator
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan akun administrator aplikasi
<b>Pre Condition</b>	Administrator sudah terautentifikasi
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Administrator memasukan data administrator terbaru.</li> <li>2. Adminsitrator melakukan perintah tambah admin.</li> <li>3. Sistem memvalidasi input administrator baru.</li> <li>4. Sistem menyimpan akun administrator baru ke dalam basis data.</li> </ol>
<b>Exception</b>	<ol style="list-style-type: none"> <li>3a. Sistem gagal melakukan validasi.</li> <li>3a. 1. Sistem menampilkan pesan kesalahan</li> <li>3a. 2. Pengguna kembali ke proses 2</li> </ol>

<b>Post Condition</b>	Administrator baru telah ditambahkan
-----------------------	--------------------------------------

e. Hapus Admin

Tabel 40. Skenario *Use Case* Hapus Admin

<b>Nama Use Case</b>	Hapus Admin
<b>Aktor</b>	Administrator
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan penghapusan akun administrator aplikasi
<b>Pre Condition</b>	Administrator sudah terautentifikasi
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Administrator memilih data administrator.</li> <li>2. Adminsitrator melakukan perintah hapus administrator.</li> <li>3. Sistem memvalidasi administrator yang akan dihapus.</li> <li>4. Sistem menghapus administrator dari basis data.</li> </ol>
<b>Exception</b>	<ol style="list-style-type: none"> <li>3a. Sistem gagal melakukan validasi.</li> <li>3a. 1. Sistem menampilkan pesan kesalahan</li> <li>3a. 2. Pengguna kembali ke proses 1</li> </ol>
<b>Post Condition</b>	Administrator yang dipilih telah terhapus

f. Sunting Password Admin

Tabel 41. Skenario *Use Case* Sunting Password Admin

<b>Nama Use Case</b>	Sunting Password Admin
<b>Aktor</b>	Administrator
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan penggantian <i>password</i> akun administrator aplikasi
<b>Pre Condition</b>	Administrator sudah terautentifikasi
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Administrator memilih data administrator.</li> <li>2. Adminsitrator melakukan perintah ganti password.</li> <li>3. Sistem menampilkan form penggantian password.</li> </ol>

	4. Administrator memasukkan password baru dan melakukan perintah simpan. 5. Sistem memvalidasi masukan password baru. 6. Sistem menghapus administrator dari basis data.
<b>Exception</b>	5a. Sistem gagal melakukan validasi. 5a. 1. Sistem menampilkan pesan kesalahan 5a. 2. Pengguna kembali ke proses 1
<b>Post Condition</b>	Password administrator yang dipilih telah terganti

g. Tambah Prodi

Tabel 42. Skenario *Use Case* Tambah Prodi

<b>Nama Use Case</b>	Tambah Prodi
<b>Aktor</b>	Administrator
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan penambahan data program studi yang disediakan oleh aplikasi
<b>Pre Condition</b>	Administrator sudah terautentifikasi
<b>Main Flow</b>	1. Administrator memasukkan data program studi baru. 2. Adminsitrator melakukan perintah tambah program studi. 3. Sistem memvalidasi input program studi baru. 4. Sistem menyimpan ke dalam basis data.
<b>Exception</b>	3a. Sistem gagal melakukan validasi. 3a. 1. Sistem menampilkan pesan kesalahan 3a. 2. Pengguna kembali ke proses 2
<b>Post Condition</b>	Program studi baru Telah ditambahkan

h. Sunting Prodi

Tabel 43. Skenario *Use Case* Sunting Prodi

<b>Nama Use Case</b>	Sunting Prodi
<b>Aktor</b>	Administrator

<b>Deskripsi</b>	<i>Use case</i> untuk melakukan penyuntingan data program studi yang disediakan oleh aplikasi
<b>Pre Condition</b>	Administrator sudah terautentifikasi
<b>Main Flow</b>	1. Sistem menyediakan <i>form</i> sunting program studi. 2. Administrator memasukkan penyuntingan data program studi dan melakukan perintah simpan data. 3. Sistem memvalidasi masukan. 4. Sistem menyimpan perubahan ke basis data.
<b>Exception</b>	3a. Sistem gagal melakukan validasi. 3a. 1. Sistem menampilkan pesan kesalahan 3a. 2. Pengguna kembali ke proses 2
<b>Post Condition</b>	Program Studi tersunting

i. Hapus Prodi

Tabel 44. Skenario *Use Case* Hapus Prodi

<b>Nama Use Case</b>	Hapus Prodi
<b>Aktor</b>	Administrator
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan penghapusan data program studi yang disediakan oleh aplikasi
<b>Pre Condition</b>	Administrator sudah terautentifikasi
<b>Main Flow</b>	1. Administrator memilih data program studi. 2. Adminsitrator melakukan perintah hapus program studi. 3. Sistem memvalidasi program studi yang akan dihapus. 4. Sistem menghapus program studi dari basis data.
<b>Exception</b>	3a. Sistem gagal melakukan validasi. 3a. 1. Sistem menampilkan pesan kesalahan 3a. 2. Pengguna kembali ke proses 1
<b>Post Condition</b>	Program studi yang dipilih telah terhapus

j. Moderasi Pengguna

Tabel 45. Skenario *Use Case* Moderasi Pengguna

<b>Nama Use Case</b>	Moderasi Pengguna
<b>Aktor</b>	Administrator
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan pemblokiran dan pengembalian akun pengguna (mahasiswa) berdasarkan laporan yang telah masuk dari pengguna
<b>Pre Condition</b>	1. Administrator sudah terautentifikasi 2. Laporan masuk mengenai karya
<b>Main Flow</b>	1. Administrator memilih karya yang akan dimoderasi. 2. Administrator memilih blokir pengguna jika pengguna belum terblokir. 3. Sistem memblokir pengguna.
<b>Exception</b>	2a. Administrator memilih kembalikan pengguna jika pengguna terblokir. 2a. 1. Sistem mengembalikan pengguna.
<b>Post Condition</b>	Akun mahasiswa terblokir atau kembali

k. Hapus Laporan

Tabel 46. Skenario *Use Case* Hapus Laporan

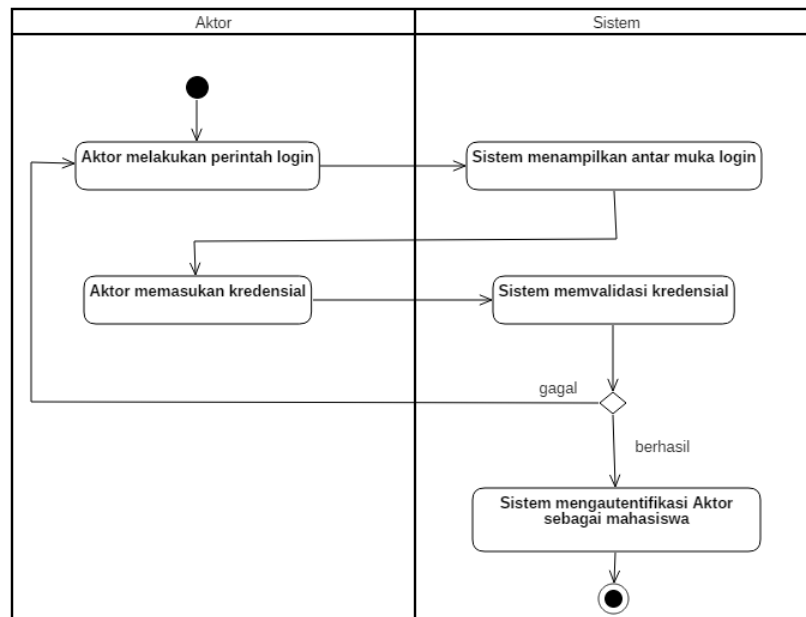
<b>Nama Use Case</b>	Hapus Laporan
<b>Aktor</b>	Administrator
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan penghapusan laporan yang masuk dari pengguna
<b>Pre Condition</b>	1. Administrator sudah terautentifikasi 2. Laporan masuk mengenai karya
<b>Main Flow</b>	1. Administrator memilih data laporan masuk. 2. Adminsitrator melakukan perintah hapus laporan masuk.



	<p>3. Sistem memvalidasi laporan masuk yang akan dihapus.</p> <p>4. Sistem menghapus laporan masuk dari basis data.</p>
<b>Exception</b>	<p>3a. Sistem gagal melakukan validasi laporan masuk.</p> <p>3a. 1. Sistem menampilkan pesan kesalahan.</p> <p>3a. 2. Pengguna kembali ke proses 1.</p>
<b>Post Condition</b>	Laporan masuk yang dipilih telah terhapus

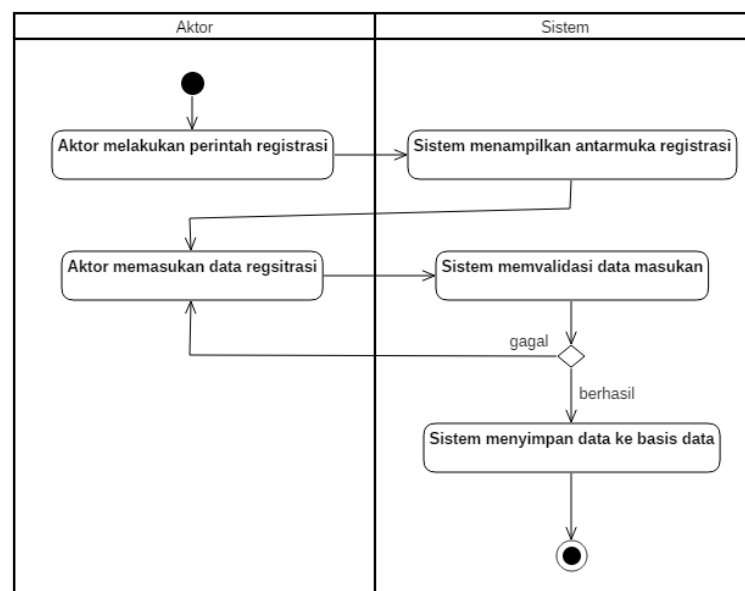
## Lampiran 6. Activity Diagram

### 1. Activity Diagram Login



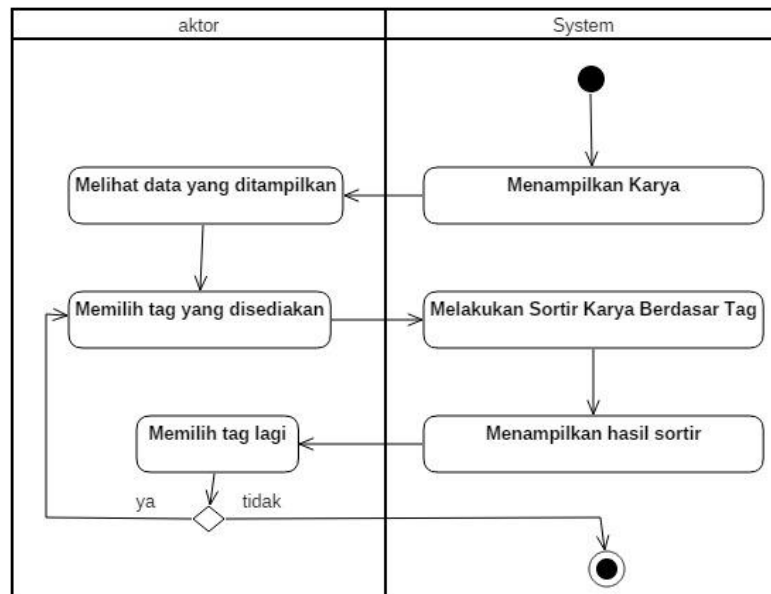
Gambar 41. Activity Diagram Login

### 2. Activity Diagram Register



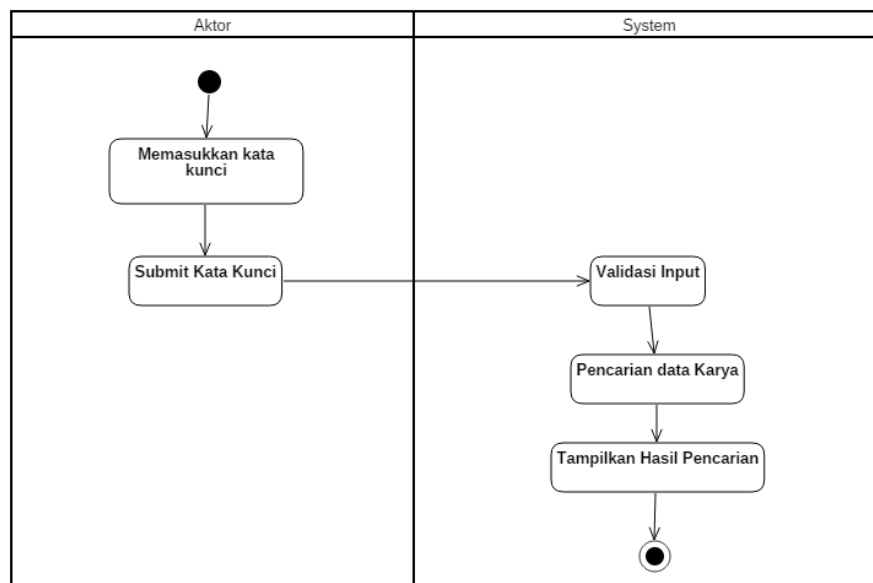
Gambar 42. Activity Diagram Register

### 3. Activity Diagram Jelajah Karya



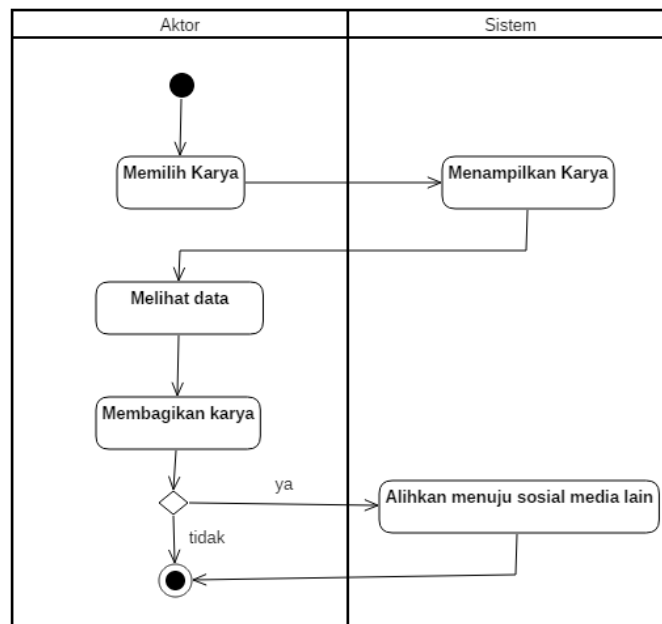
Gambar 43. Activity Diagram Jelajah Karya

### 4. Activity Diagram Pencarian Karya



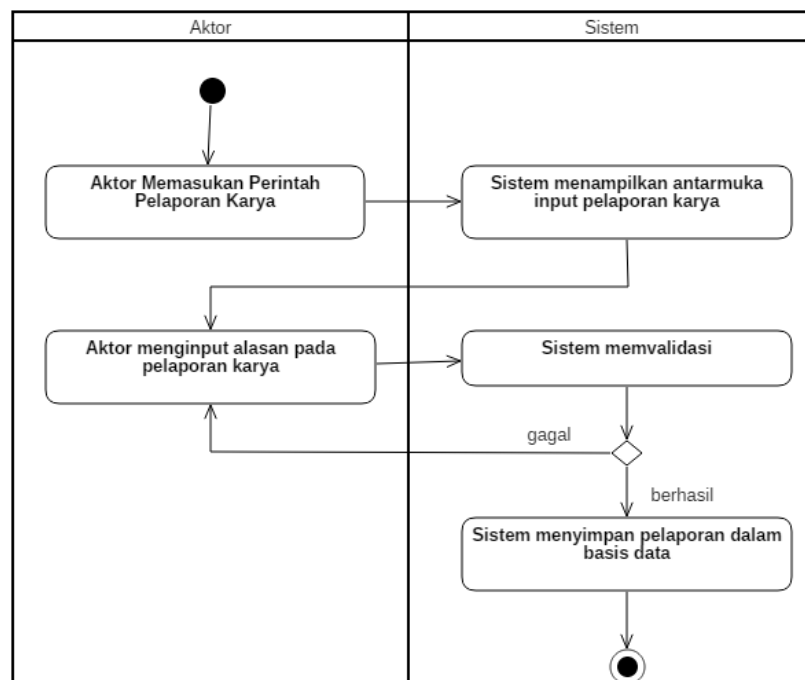
Gambar 44. Activity Diagram Pencarian Karya

## 5. *Activity Diagram* Lihat Karya



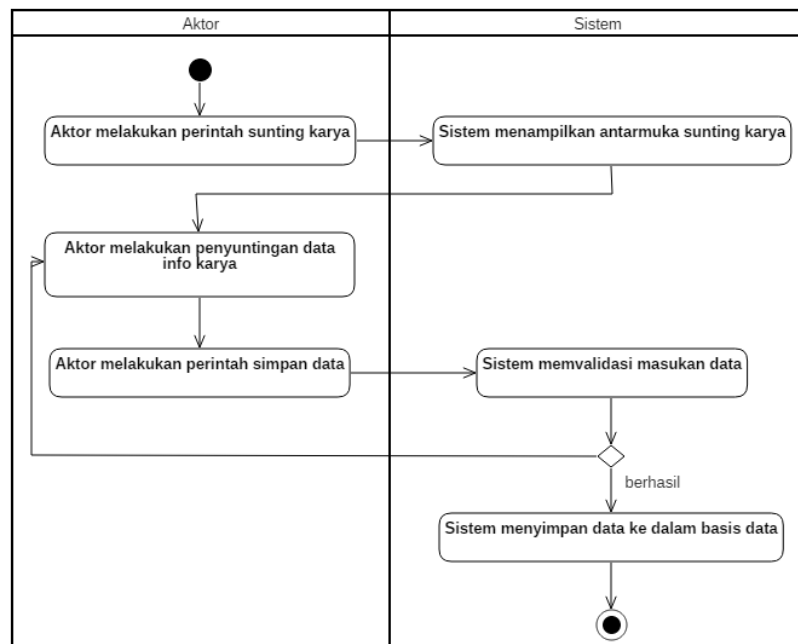
Gambar 45. *Activity Diagram* Lihat Karya

## 6. *Activity Diagram* Laporkan Karya



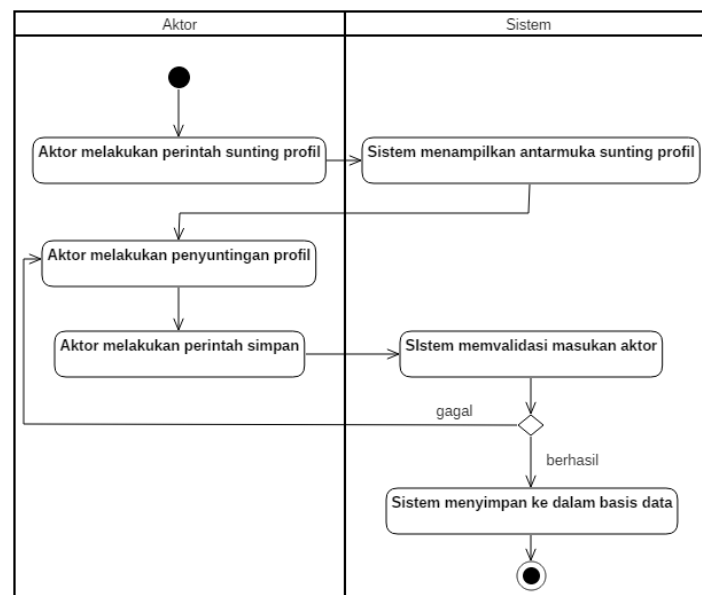
Gambar 46. *Activity Diagram* Laporkan Karya

## 7. Activity Diagram Sunting Informasi Karya



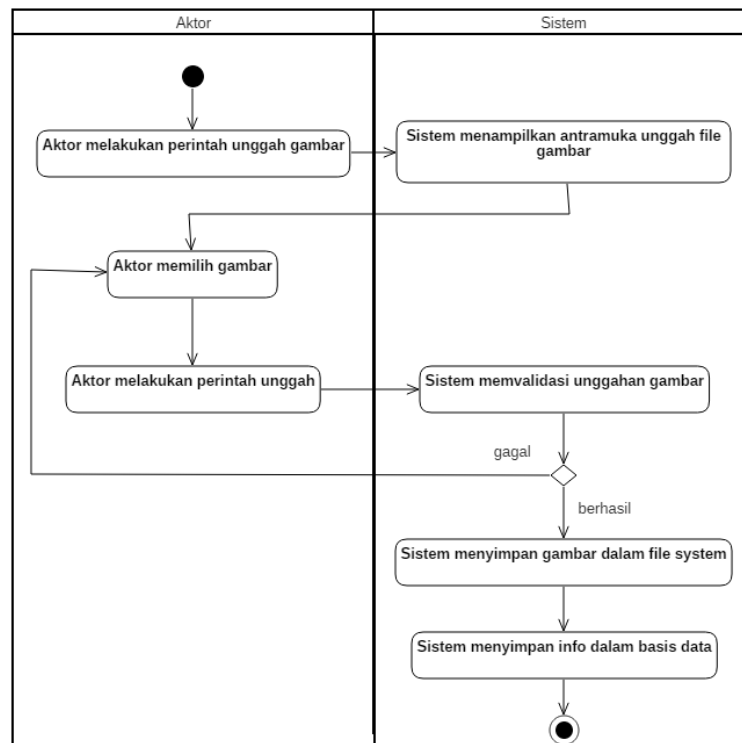
Gambar 47. Activity Diagram Sunting Informasi Karya

## 8. Activity Diagram Sunting Informasi Profil



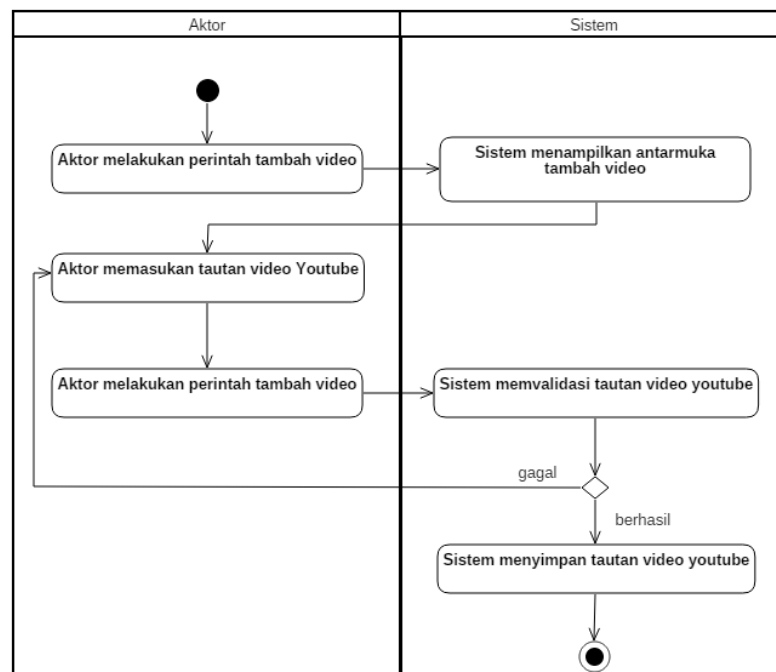
Gambar 48. Activity Diagram Sunting Informasi Profil

## 9. Activity Diagram Unggah Gambar



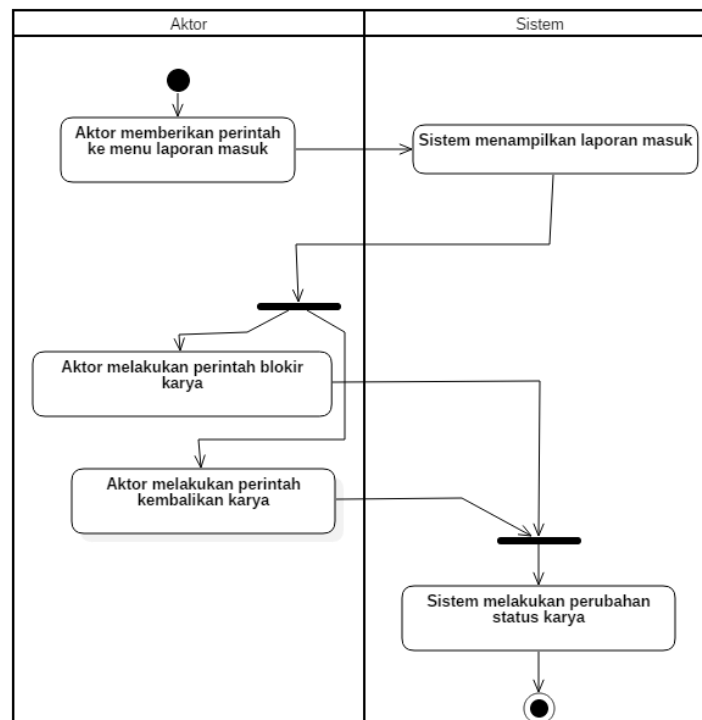
Gambar 49. Activity Diagram Unggah Gambar

## 10. Activity Diagram Tambah Video



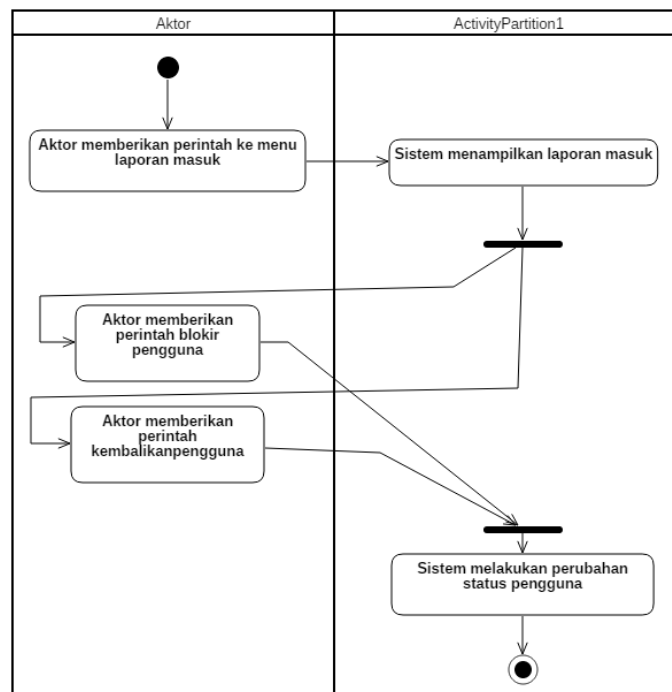
Gambar 50. Activity Diagram Tambah Video

## 11. Activity Diagram Moderasi Karya



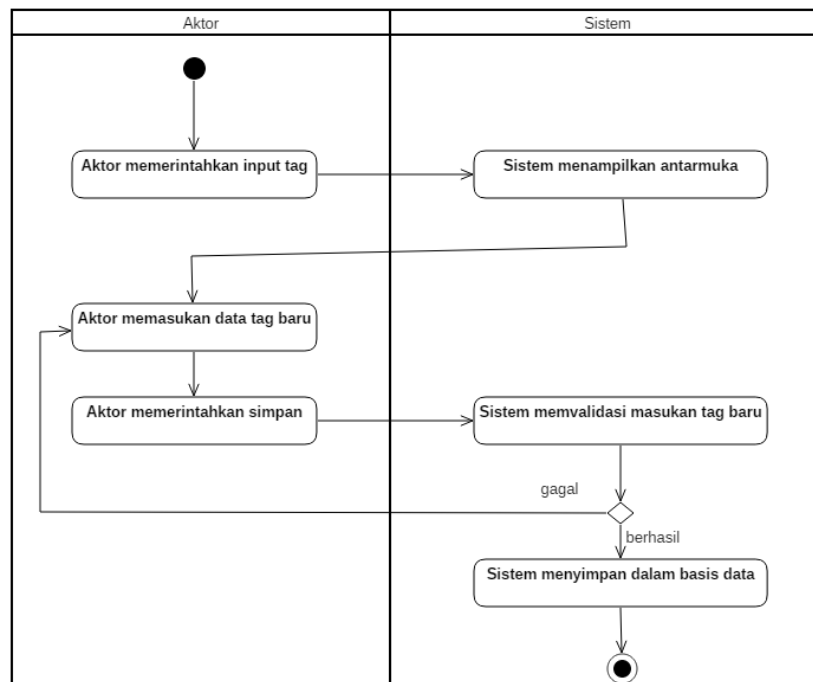
Gambar 51. Activity Diagram Moderasi Karya

## 12. Activity Diagram Moderasi Pengguna



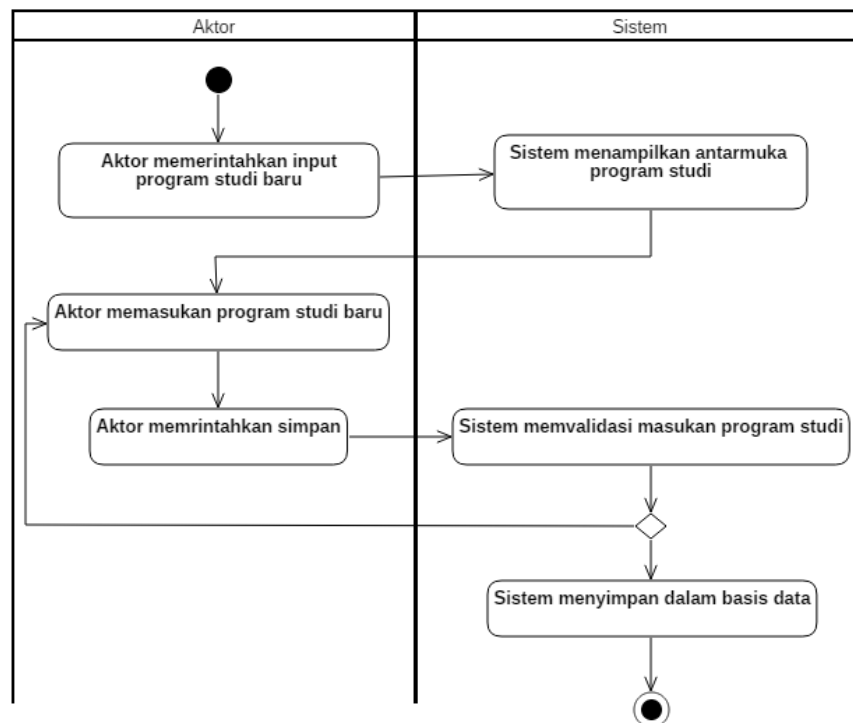
Gambar 52. Activity Diagram Moderasi Pengguna

### 13. Activity Diagram Tambah Tags



Gambar 53. Activity Diagram Tambah Tags

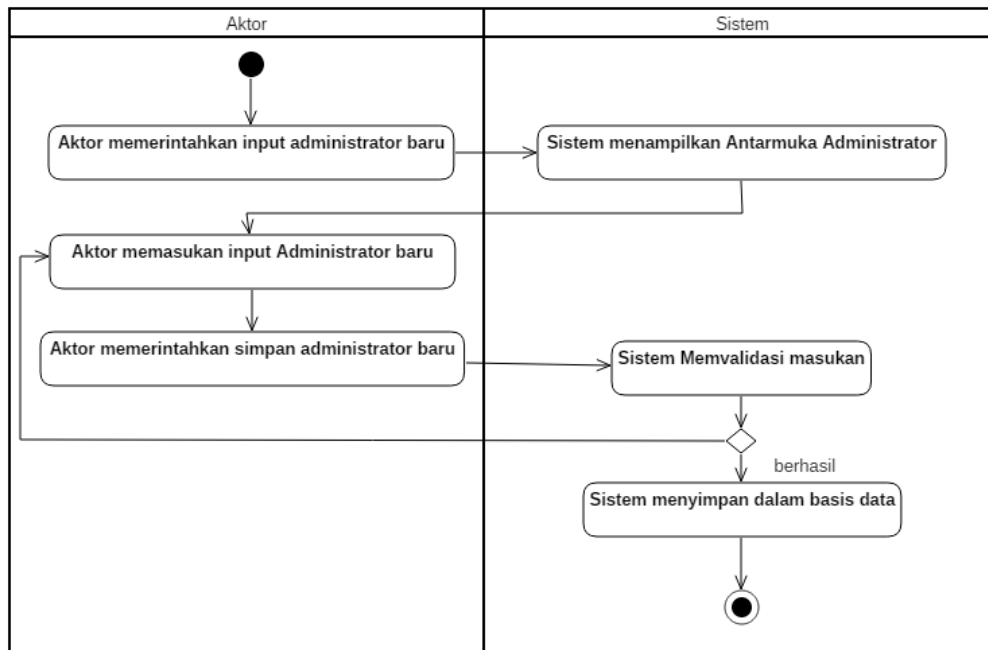
### 14. Activity Diagram Tambah Prodi



Gambar 54. Activity Diagram Tambah Prodi

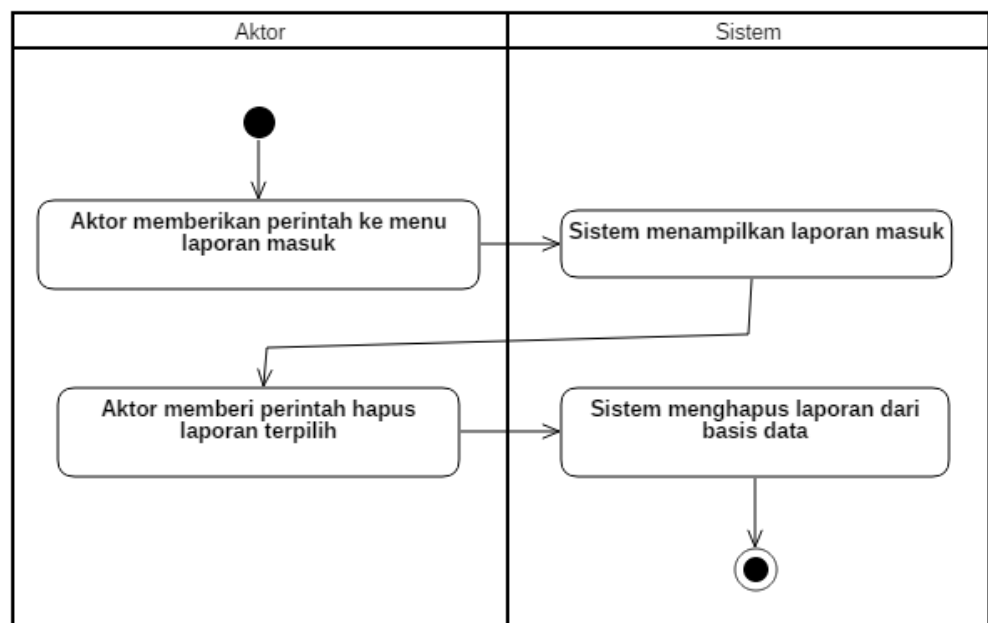


### 15. Activity Diagram Tambah Admin



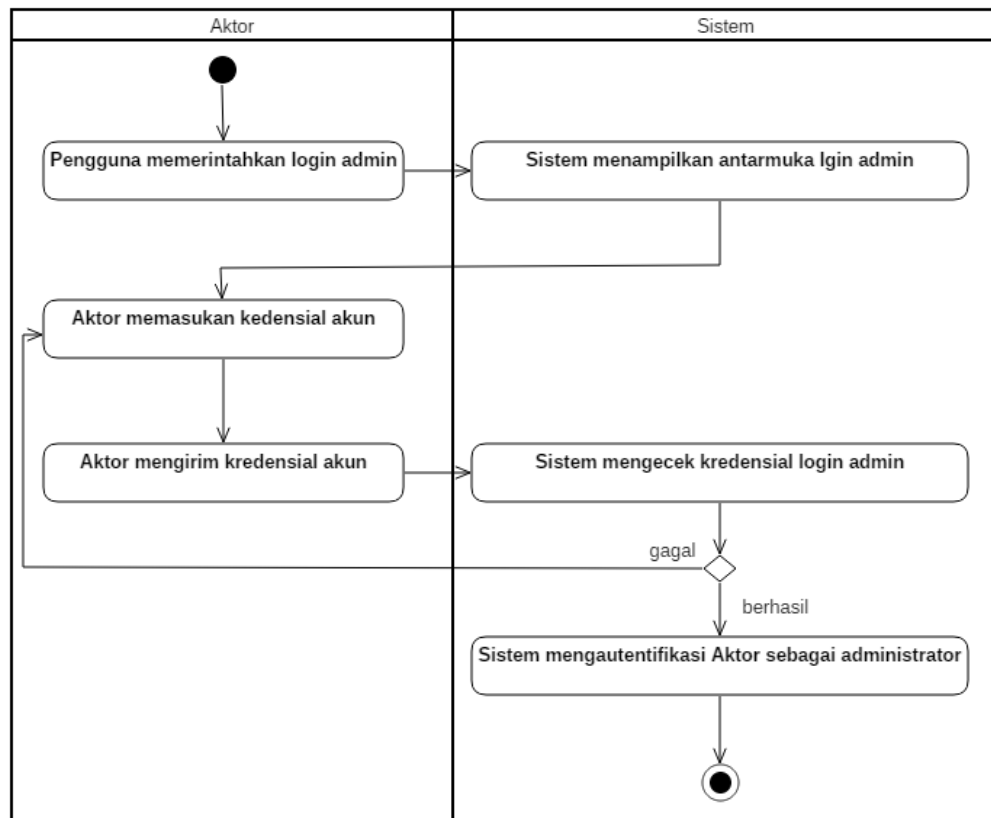
Gambar 55. Activity Diagram Tambah Admin

### 16. Activity Diagram Hapus Laporan



Gambar 56. Activity Diagram Hapus Laporan

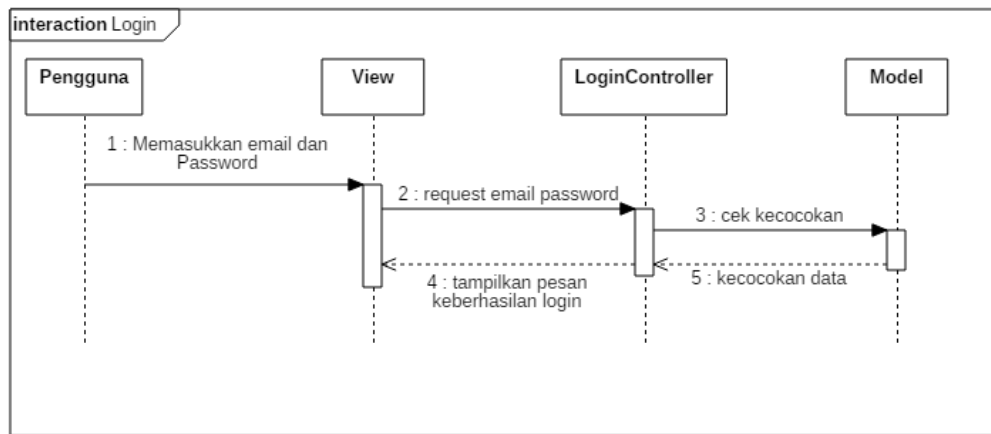
### 17. Activity Diagram Login Admin



Gambar 57. Activity Diagram Login Admin

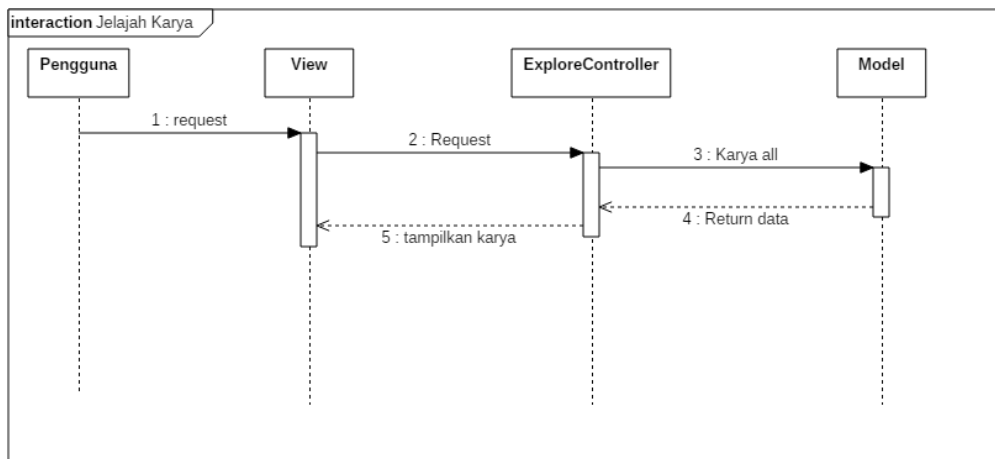
## Lampiran 7. *Sequence Diagram*

### 1. *Sequence Diagram* Login



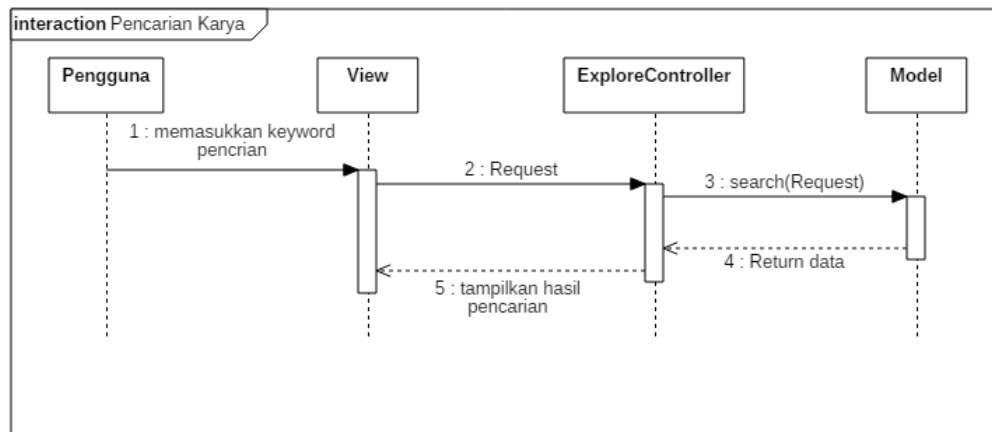
Gambar 58. *Sequence Diagram* Login

### 2. *Sequence Diagram* Jelajah Karya



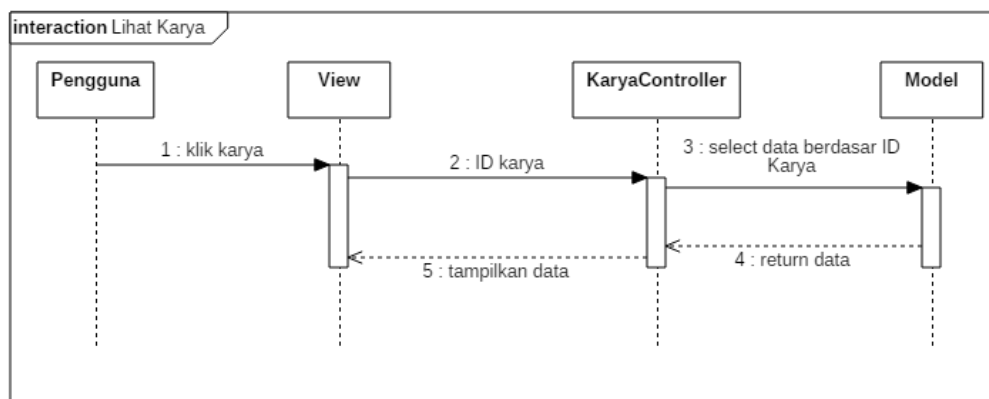
Gambar 59. *Sequence Diagram* Jelajah Karya

### 3. *Sequence Diagram* Pencarian Karya



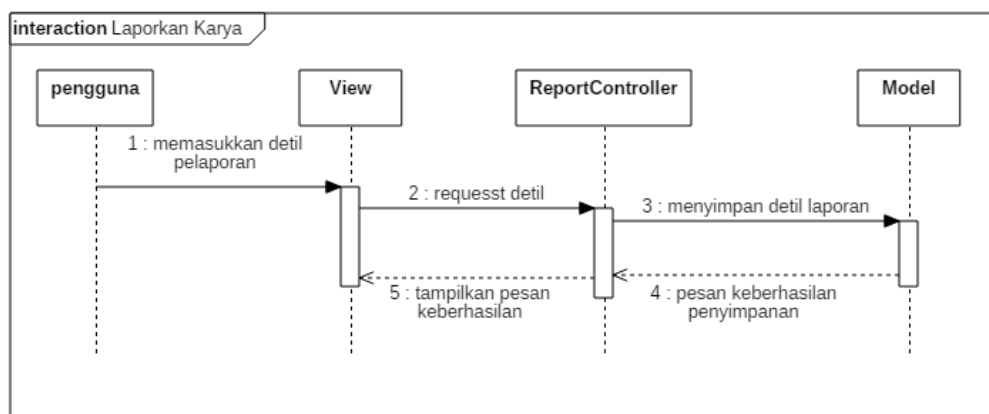
Gambar 60. *Sequence Diagram* Pencarian Karya

### 4. *Sequence Diagram* Lihat Karya



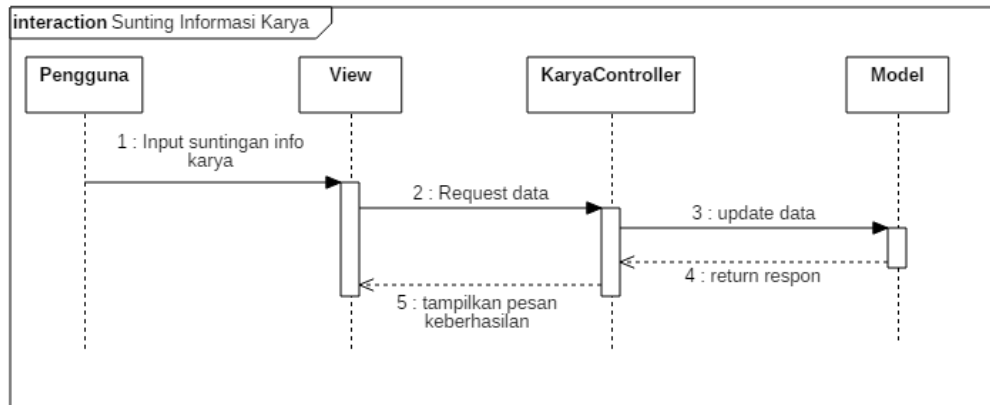
Gambar 61. *Sequence Diagram* Lihat Karya

### 5. *Sequence Diagram* Laporkan Karya



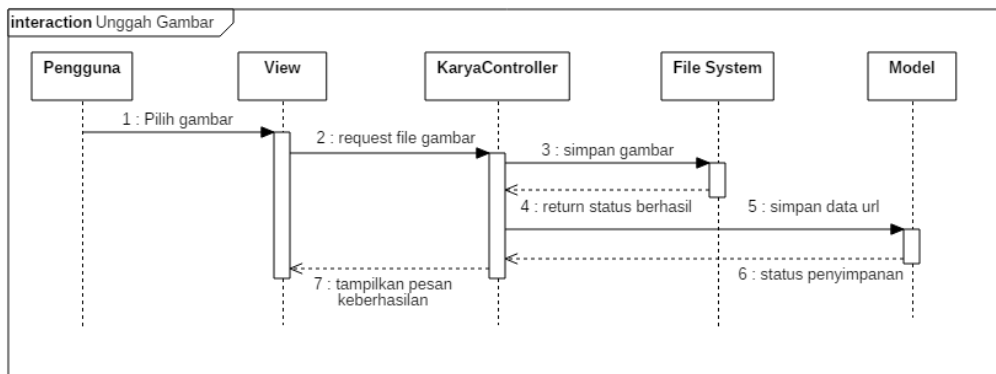
Gambar 62. *Sequence Diagram* Laporkan Karya

## 6. *Sequence Diagram* Sunting Informasi Karya



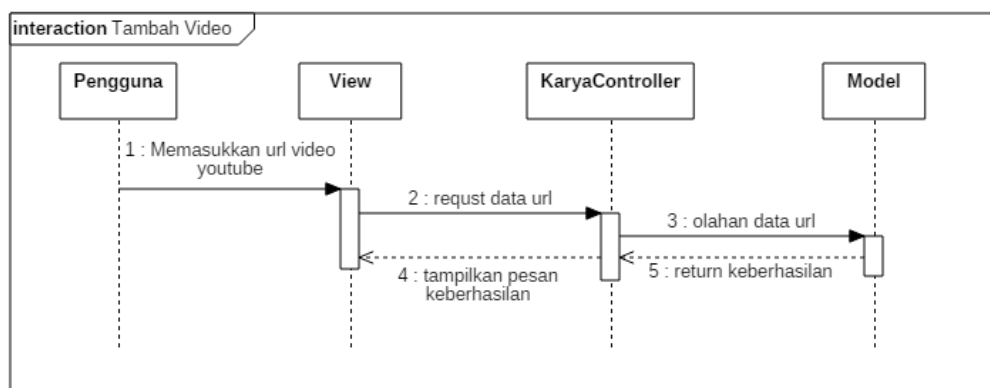
Gambar 63. *Sequence Diagram* Sunting Informasi Karya

## 7. *Sequence Diagram* Unggah Gambar



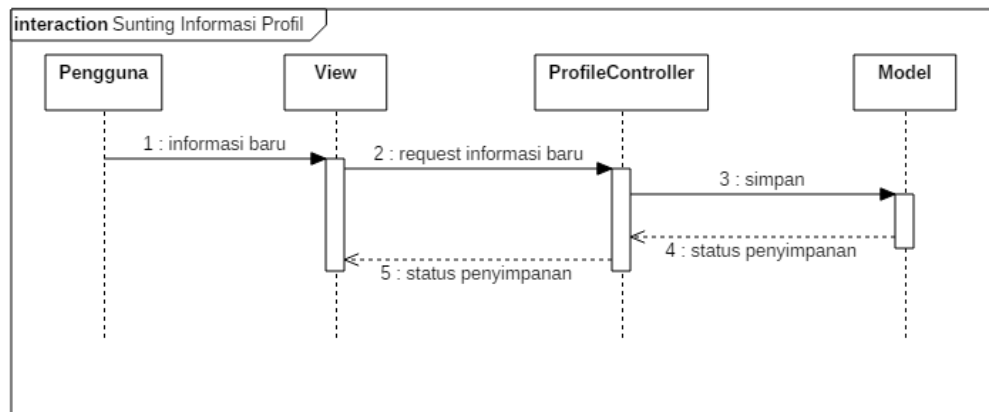
Gambar 64. *Sequence Diagram* Unggah Gambar

## 8. *Sequence Diagram* Tambah Video



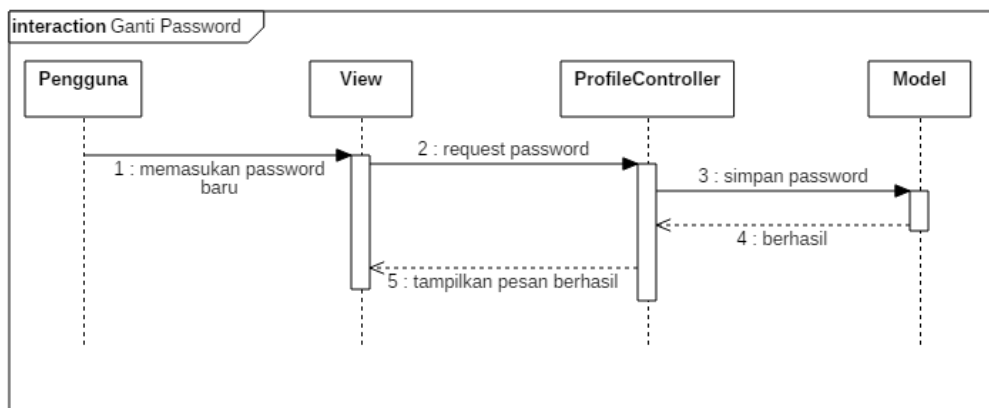
Gambar 65. *Sequence Diagram* Tambah Video

## 9. *Sequence Diagram* Sunting Informasi Profil



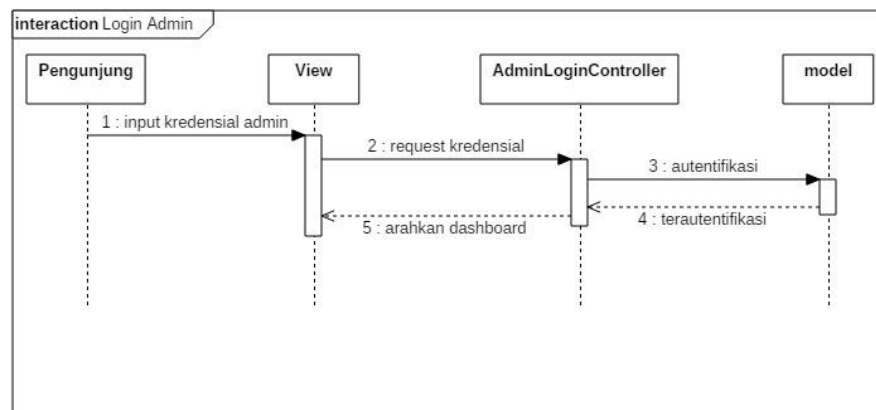
Gambar 66. *Sequence Diagram* Sunting Informasi Profil

## 10. *Sequence Diagram* Ganti Password



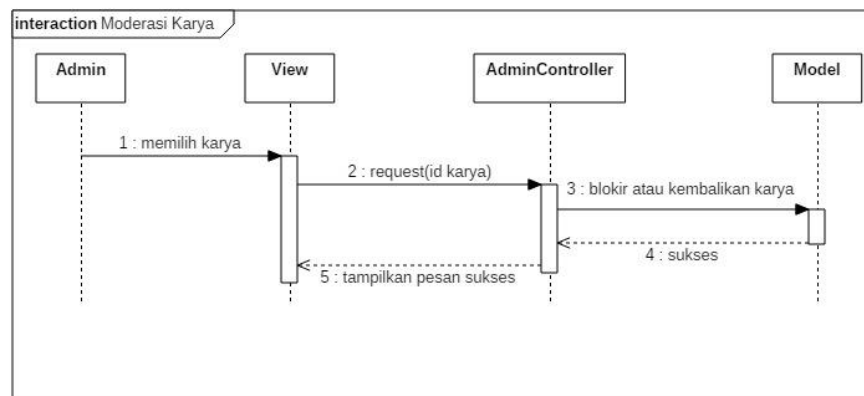
Gambar 67. *Sequence Diagram* Ganti Password

### 11. *Sequence Diagram* Login Admin



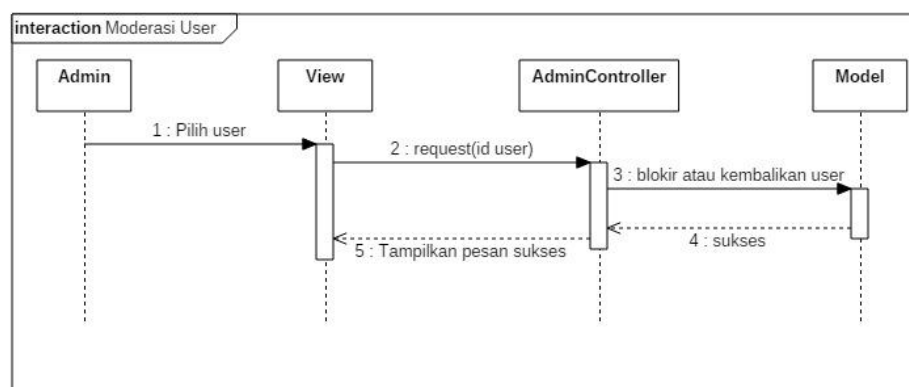
Gambar 68. *Sequence Diagram* Login Admin

### 12. *Sequence Diagram* Moderasi Karya



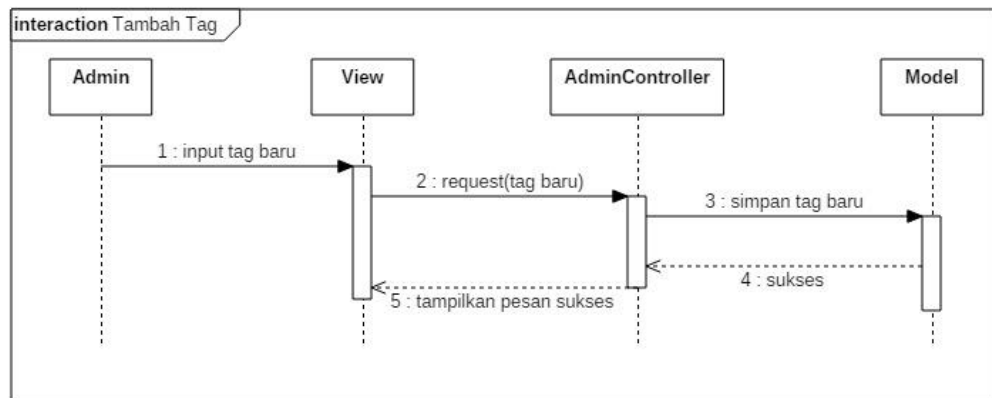
Gambar 69. *Sequence Diagram* Moderasi Karya

### 13. *Sequence Diagram* Moderasi Pengguna



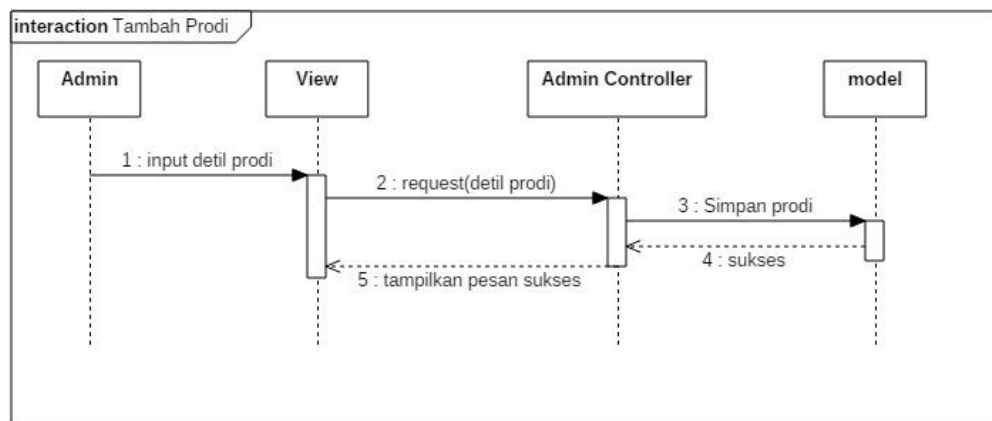
Gambar 70. *Sequence Diagram* Moderasi Pengguna

#### 14. *Sequence Diagram* Tambah Tag



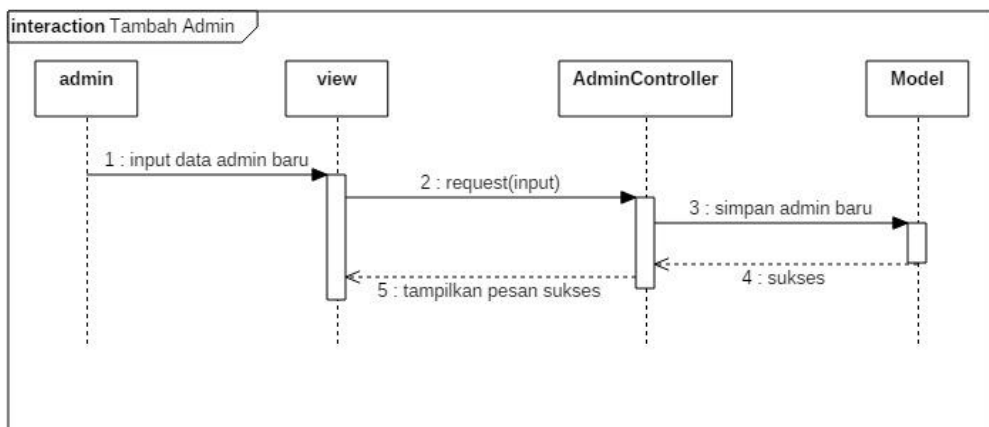
Gambar 71. *Sequence Diagram* Tambah Tag

#### 15. *Sequence Diagram* Tambah Prodi



Gambar 72. *Sequence Diagram* Tambah Prodi

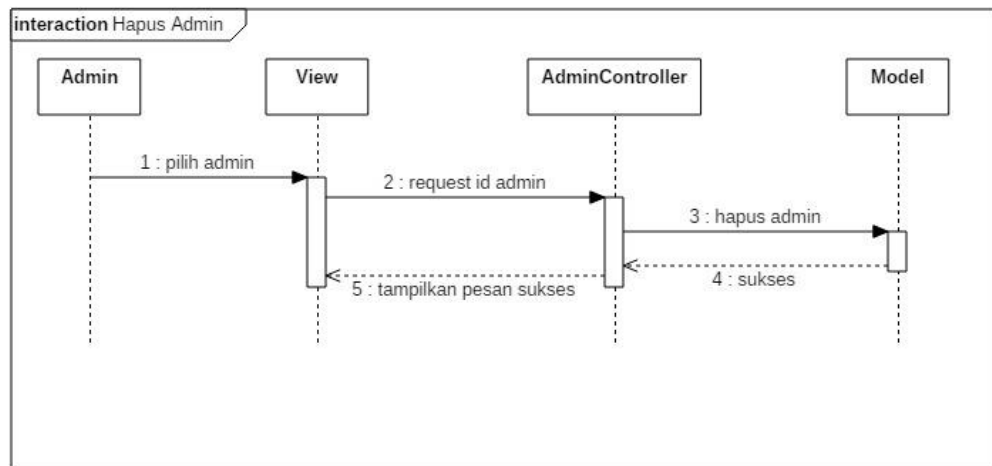
#### 16. *Sequence Diagram* Tambah Admin



Gambar 73. *Sequence Diagram* Tambah Admin



### 17. *Sequence Diagram* Hapus Admin



Gambar 74. *Sequence Diagram* Hapus Admin

## Lampiran 8. Desain (*wireframe*) Antarmuka Aplikasi.

### 1. Desain *wireframe* Halaman Register

Karyaku

52 x 50

Karyaku

Pendaftaran

Alamat Surel

bintang.muhammad@student.uny.ac.id

Kata Kunci

\*\*\*\*\*

Nama Lengkap

Bintang Muhammad

NIM

14520241057

Program Studi

Pendidikan Teknik Informatika

Daftar

[Sudah punya akun? silahkan login](#)

Gambar 75. Desain Tampilan Halaman Register

### 2. Desain *wireframe* Halaman Beranda

Karyaku

text

Bintang Muhammad

14520241057

Pend. Tek Informatika

Tambah Karya Baru

75 x 77

Karyaku

Senin, 12 April 2018

Deskripsi karyaaku adalah sebagai ini itu dan itu

Selengkapnya

75 x 77

Karyaku

Senin, 12 April 2018

Deskripsi karyaaku adalah sebagai ini itu dan itu

Selengkapnya

75 x 77

Karyaku

Senin, 12 April 2018

Deskripsi karyaaku adalah sebagai ini itu dan itu

Selengkapnya

Gambar 76. Desain Tampilan Halaman Beranda

145

### 3. Desain *wireframe* Halaman Unggah karya

Karyaku

Tambah Karya Baru...

Judul Karya

Email@mail.com

Deskripsi

text

Tambah Karya

Gambar 77. Desain Tampilan Halaman Unggah Karya

### 4. Desain *wireframe* Halaman Pencarian

Karyaku

Pencarian

Hasil pencarian dengan kata kunci "sesuatu"

160 x 133

[Karya 1](#)

160 x 133

[Karya 2](#)

160 x 133

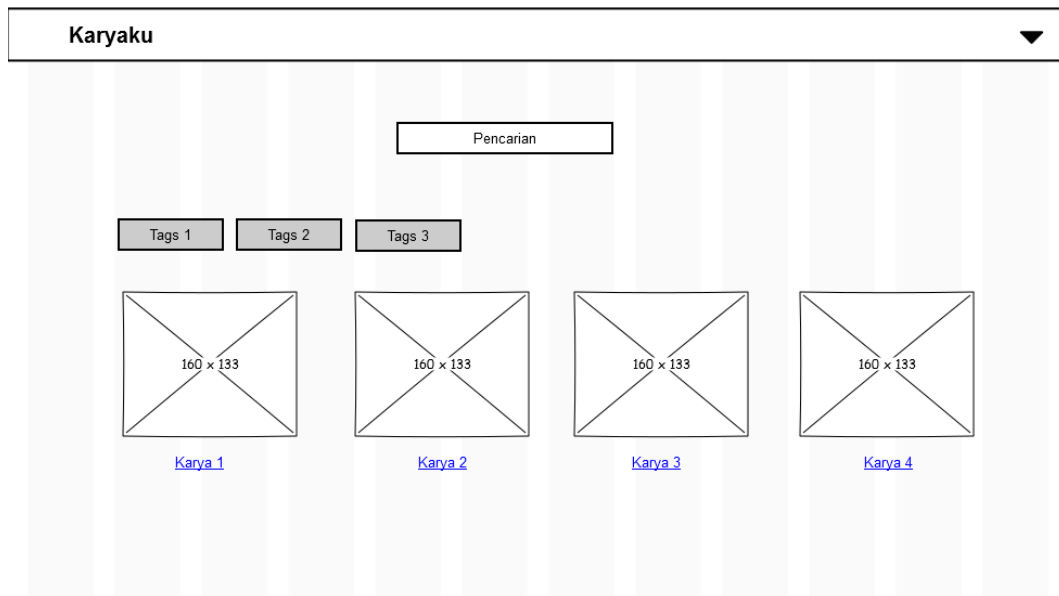
[Karya 3](#)

160 x 133

[Karya 4](#)

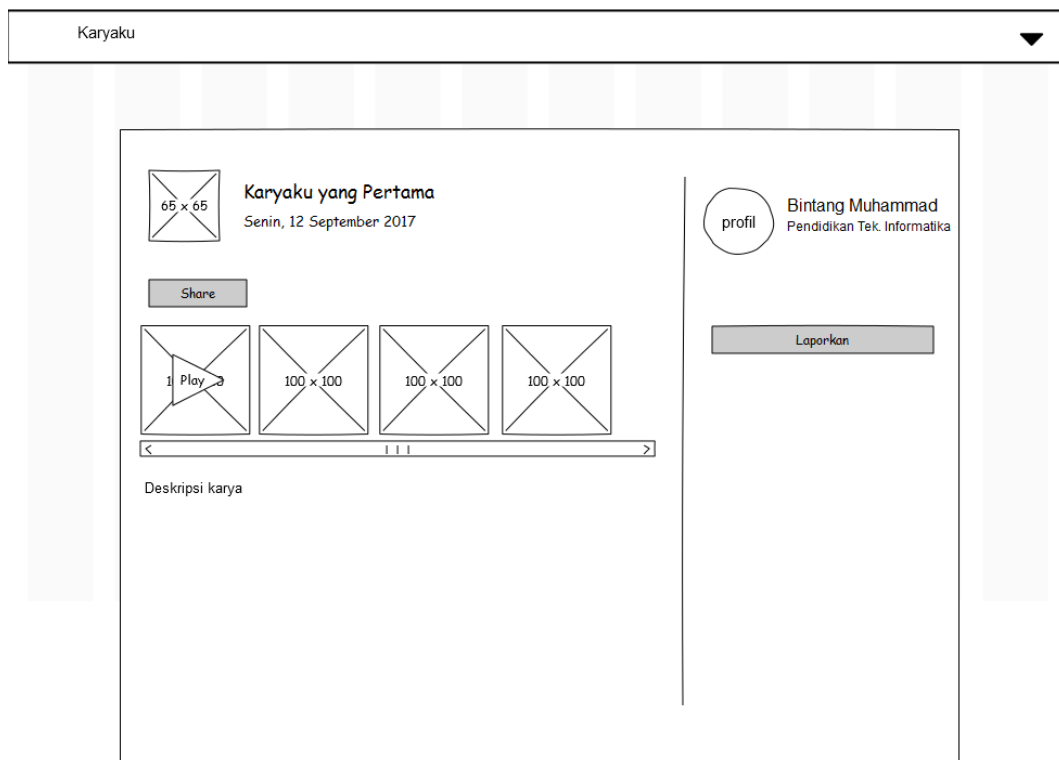
Gambar 78. Desain Tampilan Halaman Pencarian

## 5. Desain *wireframe* Halaman Jelajah



Gambar 79. . Desain Tampilan Halaman Jelajah Karya

## 6. Desain *wireframe* Halaman Karya



Gambar 80. Desain Tampilan Halaman Karya

## 7. Desain *wireframe* Halaman Sunting Karya

Karyaku

Judul Karya - sunting

[Informasi Karya](#)

[Gambar Karya](#)

[Video karya](#)

Judul

Deskripsi

Tag #1

Tag #2

Simpan Perubahan

Thumbnail

Edit Thumbs

149 x 146

Lihat Karya

Gambar 81. Desain Tampilan Halaman Sunting Karya Bagian Informasi Dasar

Karyaku

Judul Karya - sunting

[Informasi Karya](#)

[Gambar Karya](#)

[Video karya](#)

91 x 85

Pilih Gambar

Unggah gambar

Gambar yang telah diunggah

X

149 x 146

X

149 x 146

X

149 x 146

Lihat Karya

Gambar 82. Desain Tampilan Halaman Sunting Karya Bagian Galeri

148

Karyaku

Judul Karya - sunting

Lihat Karya

[Informasi Karya](#)  
[Gambar Karya](#)  
[Video karya](#)

Masukkan youtube video link

Video yang telah dicantumkan

Gambar 83. Desain Tampilan Halaman Sunting Karya Bagian Video

## 8. Desain *wireframe* Halaman Pelaporan Karya

Karyaku

Laporkan Karya

**Karya #1**  
 Bintang Muhammad

Alasan pelaporan

Gambar 84. Desain Tampilan Halaman Pelaporan Karya

## 9. Desain *wireframe* Halaman Profil

Karyaku

Profil Foto

Bintang Muhammad

14520241057

Pend. Tek. Informatika

153 x 145

Karya #1

153 x 145

Karya #2

153 x 145

Karya #3

Gambar 85. Desain Tampilan Halaman Profil

## 10. Desain *wireframe* Halaman Sunting Profil

Karyaku

Sunting Profil

Nama

NIM

Program Studi

Alamat email

Simpan Perubahan

147 x 134

Unggah Gambar Profil

Password Baru

\*\*\*\*\*

Ulangi Password

\*\*\*\*\*

Simpan Password

Gambar 86. Desain Tampilan Halaman Profil

## 11. Desain *wireframe* Halaman Login Administrator

**Karyaku**

username

\*\*\*\*\*

Login

Gambar 87. Desain Tampilan Halaman Login Administrator

## 12. Desain *wireframe* Halaman *Dashboard* Administrator

**Karyaku**

Dashboard

Laporan

Tags

Prodi

Administrator

**10 Karya**

**10 Pengguna**

**10 Prodi**

**10 Laporan**

**Karya terbaru**

MenuItem

MenuItem

MenuItem

MenuItem

MenuItem

MenuItem

MenuItem

**Lihat Semua**

**Laporan Terbaru**

MenuItem

MenuItem

MenuItem

MenuItem

MenuItem

MenuItem

MenuItem

MenuItem

MenuItem

MenuItem

MenuItem

MenuItem

**Lihat Semua**

Gambar 88. Desain Tampilan Halaman Dashboard Administrator



### 13. Desain *wireframe* Halaman Kelola Laporan Administrator

Karyaku

Dashboard

Laporan

Tags

Prodi

Administrator

Laporan Masuk

id	Karya	Pemilik	Alasan	Email Pelapor	Tanggal	Status	...
1	Karya #1	Bintang Muh.	Lorem Ipsum dolor sit amet	biforst@mail.com	23-01-2018	Blocked	X
2	Karya #2 X Block	Bintang Muh. X Block	Lorem Ipsum dolor sit amet	strombreaker@mail.com	23-01-2018	None	X

Gambar 89. Desain Tampilan Halaman Kelola Laporan

### 14. Desain *wireframe* Halaman Kelola *tags* Administrator

Karyaku

Dashboard

Laporan

Tags

Prodi

Administrator

Tag yang tersedia ...

Tag baru ...

Tambah Tag

id	Tags	Jumlah	Aksi
1	PKM	9	<div>lihat</div> <div>hapus</div>
2	Proyek Mandiri	9	<div>lihat</div> <div>hapus</div>
3	Tugas Akhir Skripsi	9	<div>lihat</div> <div>hapus</div>
4	Aplikasi Mobile	9	<div>lihat</div> <div>hapus</div>
5	Aplikasi Website	9	<div>lihat</div> <div>hapus</div>
6	Media Pembelajaran	9	<div>lihat</div> <div>hapus</div>
7	Desain Grafis	9	<div>lihat</div> <div>hapus</div>

Gambar 90. Desain Tampilan Halaman Kelola Tags

## 15. Desain *wireframe* Halaman Kelola Program Studi Administrator

Karyaku

Dashboard  
Laporan  
Tags  
Prodi  
Administrator

Program Studi

Fakultas

Tambah

id	Prodi	Fakultas	Aksi
1	Pend. Tek. Informatika	FT	lihat hapus
2	Pend. Tek. Elektronika	FT	lihat hapus
3	Pend. Tek. Otomotif	FT	lihat hapus
4	Pend. Tek. Mesin	FT	lihat hapus
5	Pend. Tek. Boga	FT	lihat hapus
6	Pend. Tek. Busana	FT	lihat hapus
7	Pend. Tek. Sipil	FT	lihat hapus
8	Pend. Akuntansi	FE	lihat hapus
9	Pend. Manajemen	FE	lihat hapus

Gambar 91. Desain Tampilan Halaman Kelola Program Studi

## 16. Desain *wireframe* Halaman Kelola Administrator

Karyaku

Dashboard  
Laporan  
Tags  
Prodi  
Administrator

Program Studi

Fakultas

Tambah

username	Nama	Email	Aksi
admin	Adminsitrator Karyaku	admin@karyaku.infiniteuny.id	lihat hapus
bintang	Bintang Muhammad	bintangmhd@gmail.com	lihat hapus

Gambar 92. Desain Tampilan Halaman Kelola Adminsitrator

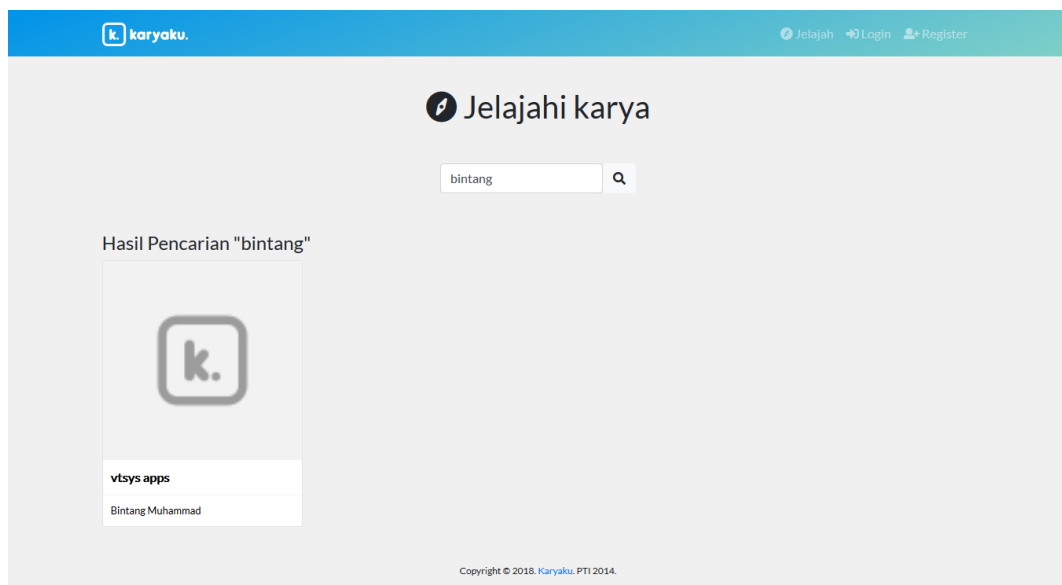
## Lampiran 9. Implementasi Antarmuka Aplikasi

### 1. Halaman Depan



Gambar 93. Implementasi Halaman Depan

### 2. Halaman Pencarian



Gambar 94. Implementasi Halaman Pencarian

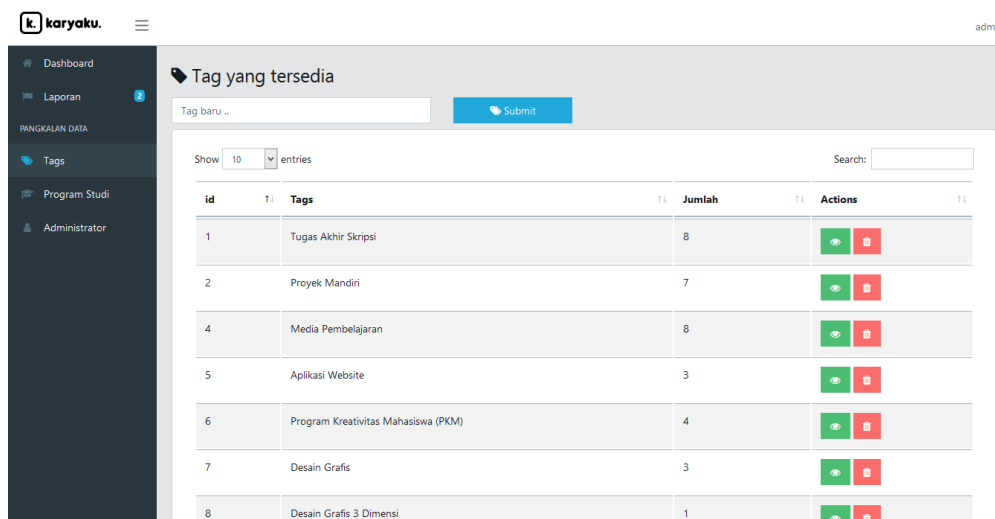
### 3. Halaman Sunting Profil

Gambar 95. Implementasi Halaman Sunting Profil

### 4. Halaman Dashboard Administrator

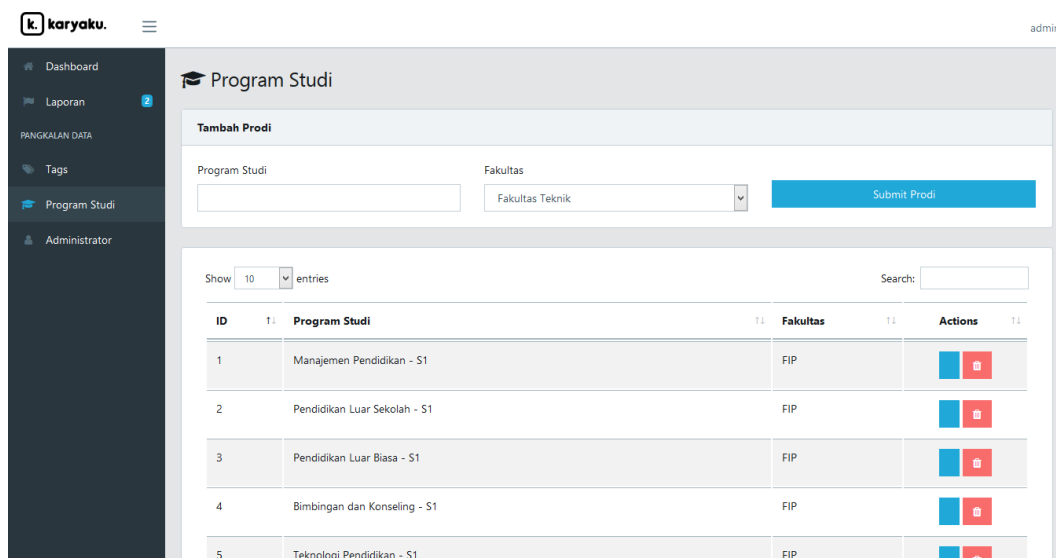
Gambar 96. Implementasi Halaman Dashboard Admin

## 5. Halaman Kelola Tags



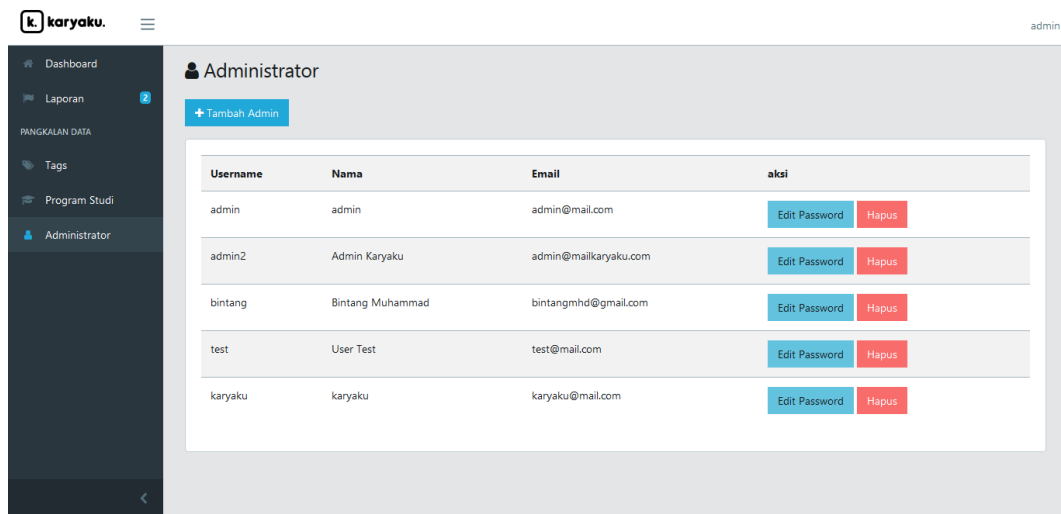
Gambar 97. Implementasi Halaman Kelola Tag

## 6. Halaman Kelola Program Studi



Gambar 98. Implementasi Halaman Kelola Program Studi

## 7. Halaman Kelola Administrator



Gambar 99. Implementasi Halaman Kelola Administrator

## Lampiran 10. Hasil Pengujian *Functional Suitability*

### Kuesioner Pengujian *Functional Suitability*

"PENGEMBANGAN APLIKASI WEB KARYAKU SEBAGAI DIREKTORI KARYA MAHASISWA  
UNIVERSITAS NEGERI YOGYAKARTA"

Nama Lengkap : Hilarius Wira Widya Iswara, S.Pd.  
Pekerjaan : IT Developer / Mahasiswa S2 Pend. Tek. Informatika  
Instansi : Wongselo.com / UNY

#### Petunjuk Pengisian:

Berilah tanda pada kolom **Ya** jika fungsi yang berjalan baik dan benar, sedangkan jika fungsi tidak berjalan dengan baik silahkan isi pada kolom **Tidak**.

No.	Fungsi	Hasil yang Diharapkan	Berhasil	
			Ya	Tidak
Umum				
Halaman Login				
1	Login	Fungsi login untuk pengguna masuk ke dalam sistem dan mengarahkan ke halaman beranda berjalan dengan benar	✓	
Halaman Register				
2	Register	Fungsi pendaftaran untuk pengguna dapat berjalan dengan benar	✓	
Halaman Jelajah				
3	Jelajah	Fungsi untuk menjelajah karya secara keseluruhan dapat berjalan dengan benar	✓	
4	Penjelahan karya berdasarkan <i>tag</i>	Fungsi untuk menjelajah karya berdasarkan <i>tag</i> karya berjalan dengan benar.	✓	
5	Pencarian karya berdasarkan kata kunci	Fungsi untuk mencari karya berdasarkan kata kunci pencarian berjalan dengan benar.	✓	
Halaman Karya				
6	Melihat karya	Fungsi untuk melihat karya dari menampilkan deskripsi, <i>thumbnail</i> , galeri, video dan pembuat karya berjalan dengan benar	✓	
7	Melaporkan karya	Fungsi untuk melaporkan karya berjalan dengan benar	✓	
8	Membagikan karya dengan tombol <i>share</i>	Fungsi untuk membagikan karya ke sosial media dapat berjalan dengan benar	✓	

No.	Fungsi	Hasil yang Diharapkan	Berhasil	
			Ya	Tidak
Pengguna				
Halaman Beranda				
9	Beranda	Fungsi untuk menampilkan beranda pengguna dan menampilkan karya-karya terbaru berjalan secara benar.	✓	
Halaman tambah karya				
10	Menambah karya baru	Fungsi untuk pengguna menambahkan karya baru dapat berjalan dengan tepat.	✓	
Halaman sunting karya				
11	Penyuntingan informasi karya	Fungsi untuk pengguna menyunting karyanya dari judul, deskripsi dan <i>tag</i> dapat berjalan dengan benar.	✓	
12	Mengelola <i>thumbnail</i> Karya	Fungsi untuk pengguna menyunting <i>thumbnail</i> karya berjalan dengan benar.	✓	
13	Mengelola gambar galeri karya	Fungsi untuk pengguna mengelola gambar galeri dari mengunggah dan menghapus dapat berjalan dengan benar.	✓	
14	Mengelola video karya	Fungsi untuk pengguna mengelola video pendukung dari menambah dan menghapus dapat berjalan dengan benar.	✓	
Halaman sunting profil				
15	Mengelola informasi dasar profil	Fungsi untuk pengguna menyunting informasi dasar dari nama, email, nim, dan prodi dapat berjalan dengan benar.	✓	
16	Mengelola gambar profil	Fungsi untuk pengguna mengunggah gambar profil dapat berjalan dengan benar.	✓	
17	Mengganti <i>password</i>	Fungsi untuk pengguna mengganti <i>password</i> dapat berjalan dengan benar.	✓	
Navigasi				
18	Logout user	Fungsi untuk pengguna keluar dari aplikasi dapat berjalan dengan benar.	✓	
Administrator				
Halaman Login				
19	Login admin	Fungsi login untuk admin ke halaman beranda admin berjalan dengan benar	✓	



No.	Fungsi	Hasil yang Diharapkan	Berhasil	
			Ya	Tidak
Halaman Laporan				
20	Mengelola laporan	Fungsi pengelolaan laporan dari melihat, menghapus, serta memblokir karya dan pengguna dapat berjalan dengan benar	✓	
Halaman <i>user</i> administrator				
21	Mengelola <i>user</i> admin	Fungsi mengelola user admin dari menambah, menghapus, dan mengganti <i>password</i> user admin dapat berjalan dengan benar.	✓	
Halaman program studi				
22	Mengelola program studi	Fungsi mengelola program studi dari menambah, menghapus, dan mengganti nama program studi dapat berjalan dengan benar.	✓	
Halaman <i>tag</i>				
23	Mengelola <i>tag</i>	Fungsi mengelola tag dari menambah dan menghapus dapat berjalan dengan benar.	✓	
Navigasi				
24	<i>Logout</i> admin	Fungsi keluar dari sistem administrator aplikasi dapat berjalan dengan benar.	✓	

**Komentar dan Saran:**

Sudah bagus, usahakan tampilan lebih responsif lagi

.....

.....

.....

Yogyakarta, 6 Juni 2018

Penguji

Hilarius Wira Widya I.

## Lampiran 11. Hasil Pengujian *Usability*

### Kuesioner Pengujian *Usability*

"PENGEMBANGAN APLIKASI WEB KARYAKU SEBAGAI DIREKTORI KARYA MAHASISWA  
UNIVERSITAS NEGERI YOGYAKARTA"

Nama : Ahmad Fahmi Asar  
Program Studi : Pendidikan Teknik Informatika  
Angkatan : 2016

#### Petunjuk Pengisian:

1. Sebelum mengisi kuesioner ini pastikan anda telah menggunakan aplikasi Karyaku.
2. Tulis identitas di tempat yang telah disediakan.
3. Bacalah pertanyaan dan beri tanda centang (✓) pada kolom yang pilihan jawaban sesuai dengan keterangan dibawah.


#### Keterangan:

SS : Sangat Setuju  
S : Setuju  
RG : Ragu-ragu  
TS : Tidak Setuju  
STS : Sangat Tidak Setuju

No.	Pertanyaan	Jawaban				
		SS	S	RG	TS	STS
Usefulness						
1	Aplikasi ini membantu saya menjadi lebih efektif		✓			
2	Aplikasi ini membantu saya mejadi lebih produktif		✓			
3	Aplikasi ini bermanfaat	✓				
4	Aplikasi ini memberikan dampak yang besar terhadap tugas saya lakukan dalam hidup saya		✓			
5	Aplikasi ini memudahkan saya dalam mencapai hal-hal yang diinginkan	✓				
6	Aplikasi ini menghemat waktu saya ketika saya menggunakannya	✓				
7	Aplikasi ini sesuai dengan kebutuhan saya		✓			
8	Aplikasi ini sesuai dengan apa yang saya harapkan			✓		
Ease of Use						
9	Aplikasi ini mudah digunakan	✓				
10	Aplikasi ini praktis untuk digunakan	✓				
11	Aplikasi ini mudah dipahami	✓				
12	Aplikasi ini hanya memerlukan langkah-langkah singkat dalam menggunakannya	✓				
13	Aplikasi ini dapat disesuaikan dengan kebutuhan	✓				
14	Tidak ada kesulitan dalam menggunakan aplikasi ini	✓				

No.	Pertanyaan	Jawaban				
		SS	S	RG	TS	STS
15	Saya dapat menggunakan tanpa instruksi tertulis	✓				
16	Saya melihat aplikasi ini sudah konsisten ketika dipergunakan		✓			
17	Baik pengguna yang jarang maupun yang rutin akan suka menggunakan aplikasi ini		✓			
18	Saya dapat kembali dari kesalahan dengan cepat dan mudah		✓			
19	Saya dapat menggunakan aplikasi ini dengan berhasil setiap saya gunakan	✓				
<b>Ease of Learning</b>						
20	Saya belajar menggunakan aplikasi ini dengan cepat	✓				
21	Saya mengingat penggunaan aplikasi ini dengan mudah	✓				
22	Penggunaan aplikasi ini mudah dipelajari	✓				
23	Saya mahir menggunakan aplikasi ini dengan cepat	✓				
<b>Satisfaction</b>						
24	Saya puas dengan aplikasi ini		✓			
25	Saya merekomendasikan aplikasi ini kepada teman-teman	✓				
26	Aplikasi ini menyenangkan untuk digunakan		✓			
27	Aplikasi ini bekerja sesuai dengan yang saya inginkan		✓			
28	Aplikasi ini sangat bagus	✓				
29	Saya merasa harus menggunakan aplikasi ini secara maksimal		✓			
30	Aplikasi ini nyaman untuk digunakan	✓				

Yogyakarta, 7 Mei 2018  
Responden

  
Ahmad Fauzan Asrar

### Kuesioner Pengujian *Usability*

"PENGEMBANGAN APLIKASI WEB KARYAKU SEBAGAI DIREKTORI KARYA MAHASISWA  
UNIVERSITAS NEGERI YOGYAKARTA"

Nama : Linda Noviasari  
Program Studi : Pendi. Teknik Elektronika  
Angkatan : 2014

#### Petunjuk Pengisian:

1. Sebelum mengisi kuesioner ini pastikan anda telah menggunakan aplikasi Karyaku.
2. Tulis identitas di tempat yang telah disediakan.
3. Bacalah pertanyaan dan beri tanda centang (✓) pada kolom yang pilihan jawaban sesuai dengan keterangan dibawah.

#### Keterangan:

SS : Sangat Setuju  
S : Setuju  
RG : Ragu-ragu  
TS : Tidak Setuju  
STS : Sangat Tidak Setuju

No.	Pertanyaan	Jawaban				
		SS	S	RG	TS	STS
Usefulness						
1	Aplikasi ini membantu saya menjadi lebih efektif	✓				
2	Aplikasi ini membantu saya mejadi lebih produktif	✓				
3	Aplikasi ini bermanfaat		✓			
4	Aplikasi ini memberikan dampak yang besar terhadap tugas saya lakukan dalam hidup saya		✓			
5	Aplikasi ini memudahkan saya dalam mencapai hal-hal yang diinginkan	✓				
6	Aplikasi ini menghemat waktu saya ketika saya menggunakannya		✓			
7	Aplikasi ini sesuai dengan kebutuhan saya		✓			
8	Aplikasi ini sesuai dengan apa yang saya harapkan	✓				
Ease of Use						
9	Aplikasi ini mudah digunakan		✓			
10	Aplikasi ini praktis untuk digunakan		✓			
11	Aplikasi ini mudah dipahami		✓			
12	Aplikasi ini hanya memerlukan langkah-langkah singkat dalam menggunakannya	✓				
13	Aplikasi ini dapat disesuaikan dengan kebutuhan	✓				
14	Tidak ada kesulitan dalam menggunakan aplikasi ini		✓			

No.	Pertanyaan	Jawaban				
		SS	S	RG	TS	STS
15	Saya dapat menggunakan tanpa instruksi tertulis		✓			
16	Saya melihat aplikasi ini sudah konsisten ketika dipergunakan		✓			
17	Baik pengguna yang jarang maupun yang rutin akan suka menggunakan aplikasi ini	✓				
18	Saya dapat kembali dari kesalahan dengan cepat dan mudah		✓			
19	Saya dapat menggunakan aplikasi ini dengan berhasil setiap saya gunakan	✓				
<b>Ease of Learning</b>						
20	Saya belajar menggunakan aplikasi ini dengan cepat	✓				
21	Saya mengingat penggunaan aplikasi ini dengan mudah		✓			
22	Penggunaan aplikasi ini mudah dipelajari	✓				
23	Saya mahir menggunakan aplikasi ini dengan cepat	✓				
<b>Satisfaction</b>						
24	Saya puas dengan aplikasi ini		✓			
25	Saya merekomendasikan aplikasi ini kepada teman-teman	✓				
26	Aplikasi ini menyenangkan untuk digunakan		✓			
27	Aplikasi ini bekerja sesuai dengan yang saya inginkan		✓			
28	Aplikasi ini sangat bagus		✓			
29	Saya merasa harus menggunakan aplikasi ini secara maksimal		✓			
30	Aplikasi ini nyaman untuk digunakan	✓				

Yogyakarta, 07 Mei 2018

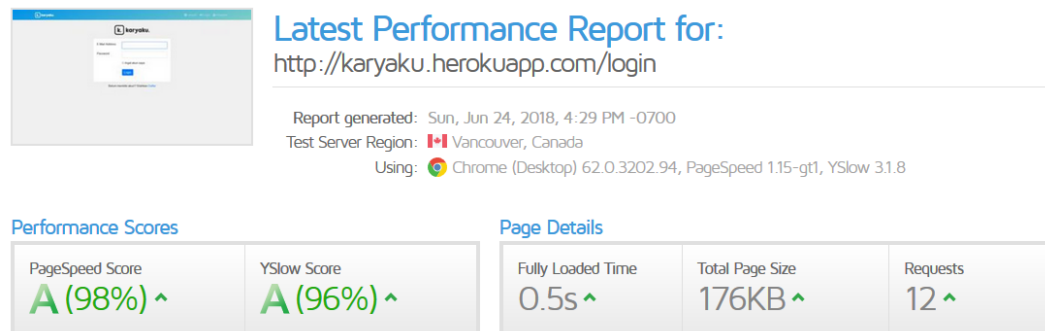
Responden



Linda Noviasari

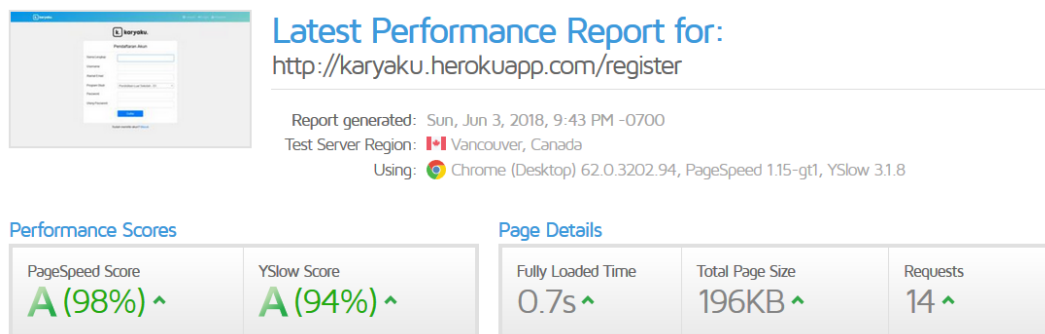
## Lampiran 12. Hasil Pengujian *Performance Efficiency*

### 1. Halaman Login



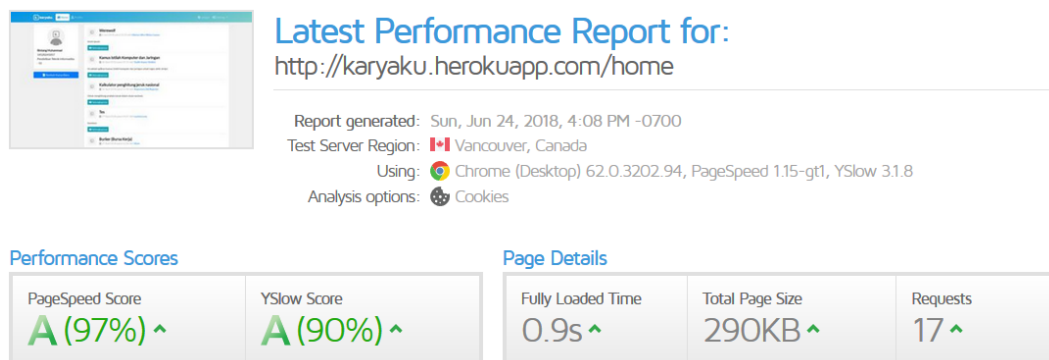
Gambar 100. Hasil Pengujian *Performance Efficiency* Halaman Login

### 2. Halaman Register



Gambar 101. Hasil Pengujian *Performance Efficiency* Halaman Register

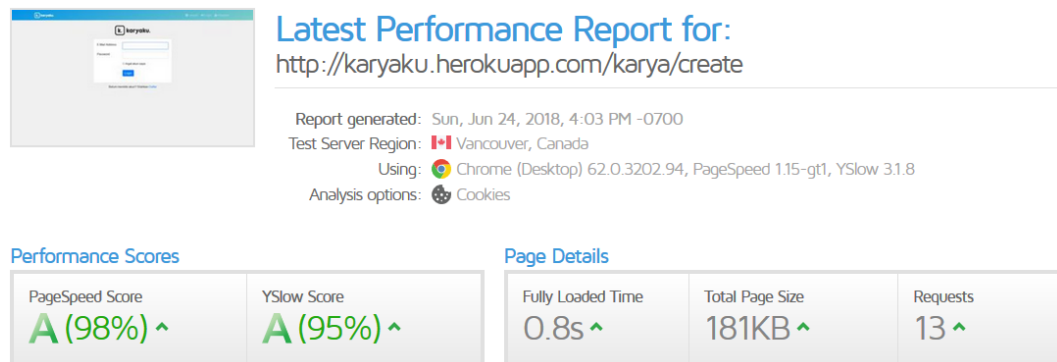
### 3. Halaman Beranda



Gambar 102. Hasil Pengujian *Performance Efficiency* Halaman Beranda

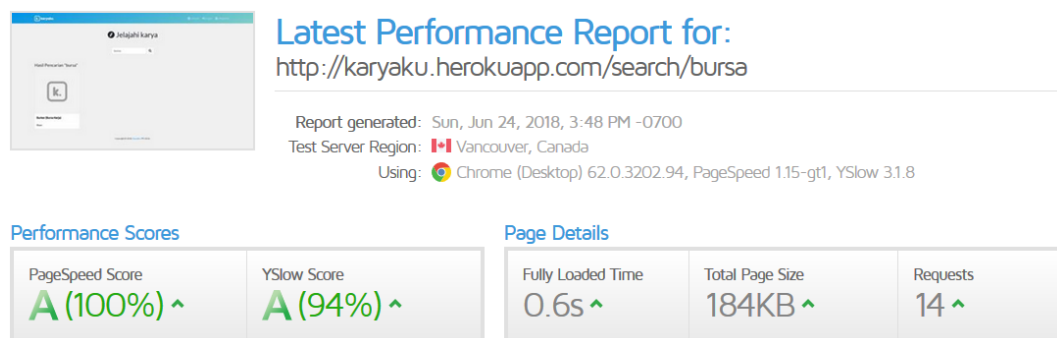


#### 4. Halaman Unggah Karya



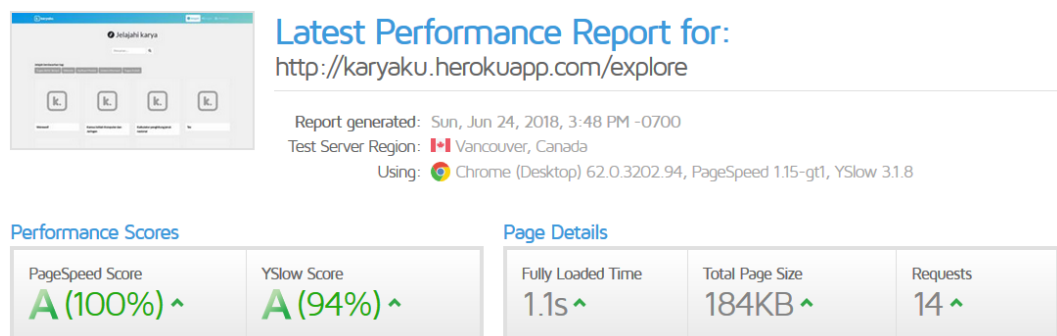
Gambar 103. Hasil Pengujian *Performance Efficiency* Halaman Unggah Karya

#### 5. Halaman Pencarian



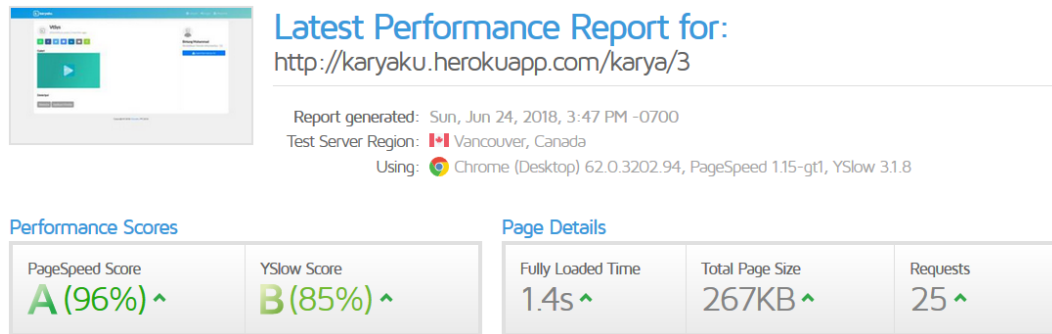
Gambar 104. Hasil Pengujian *Performance Efficiency* Halaman Pencarian

#### 6. Halaman Jelajah



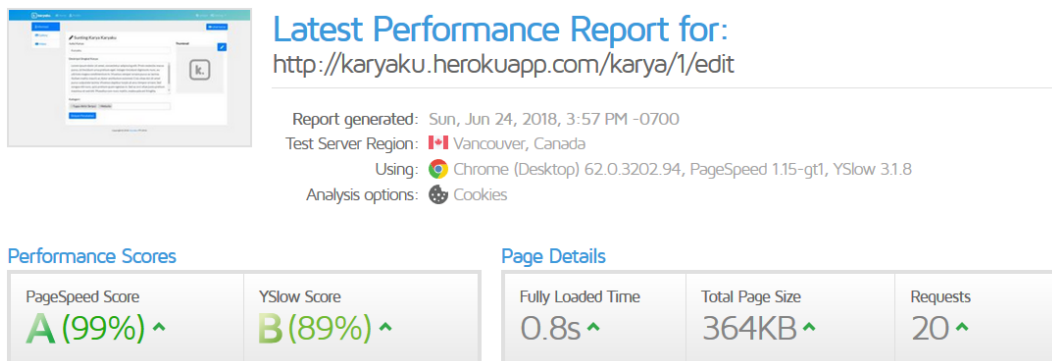
Gambar 105. Hasil Pengujian *Performance Efficiency* Halaman Jelajah

## 7. Halaman Karya



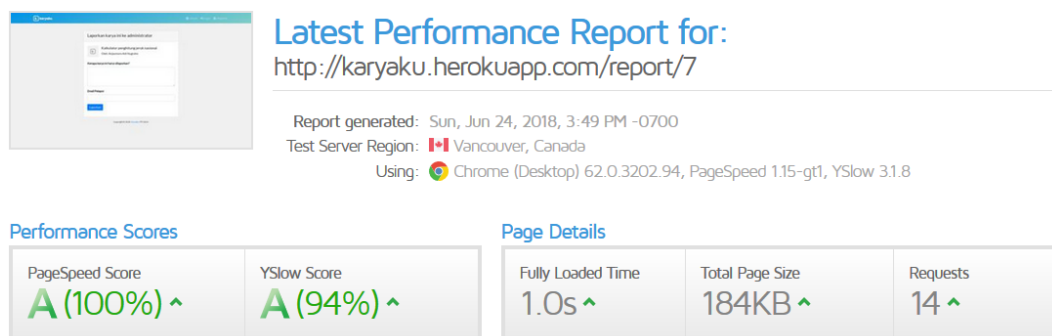
Gambar 106. Hasil Pengujian *Performance Efficiency* Halaman Karya

## 8. Halaman Sunting Karya



Gambar 107. Hasil Pengujian *Performance Efficiency* Halaman Sunting Karya

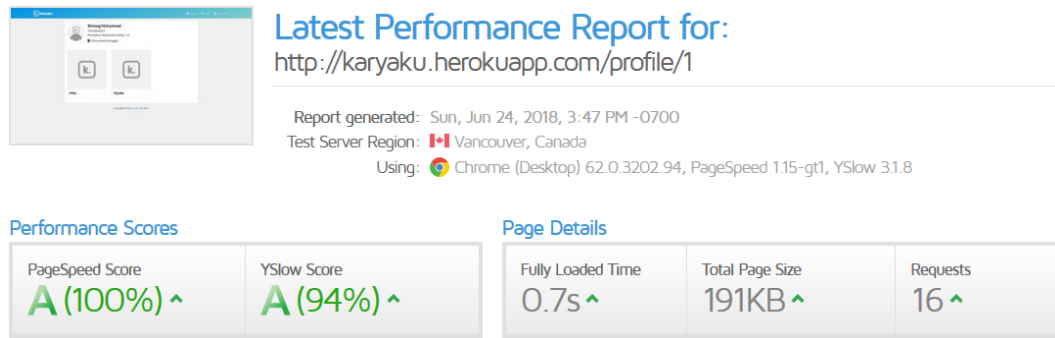
## 9. Halaman Pelaporan Karya



Gambar 108. Hasil Pengujian *Performance Efficiency* Halaman Pelaporan Karya

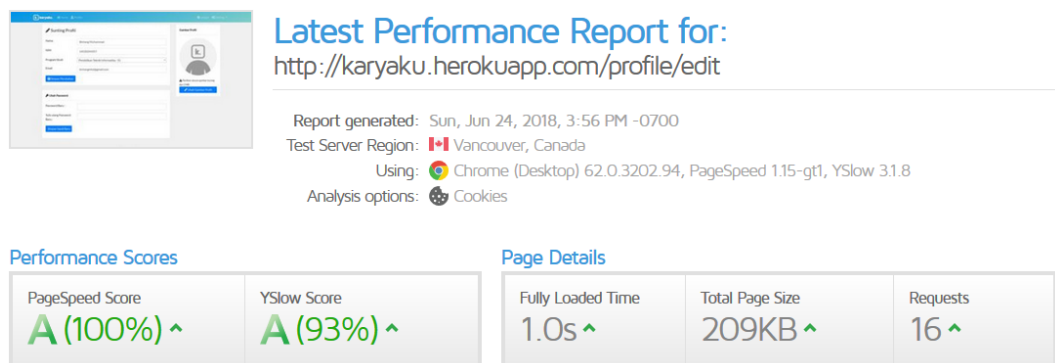
## 10. Halaman Profil





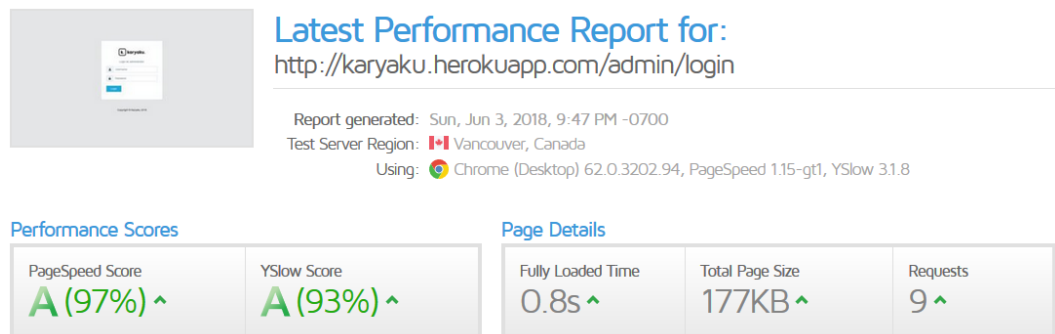
Gambar 109. Hasil Pengujian *Performance Efficiency* Halaman Profil

## 11. Halaman Sunting Profil



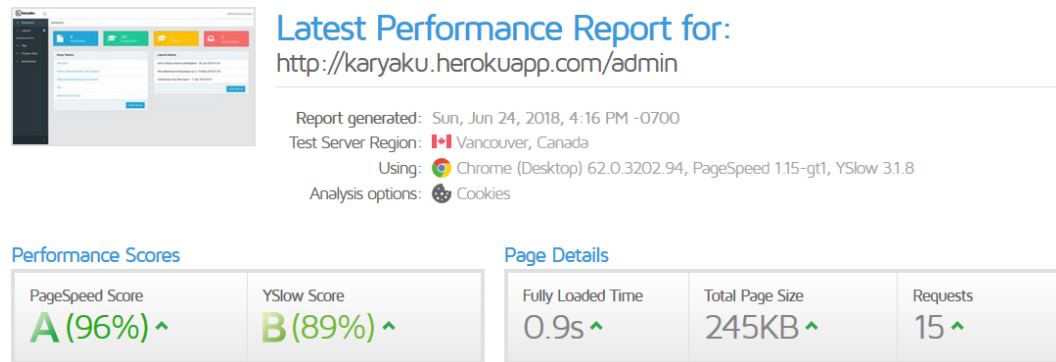
Gambar 110. Hasil Pengujian *Performance Efficiency* Halaman Sunting Profil

## 12. Halaman Login Admin



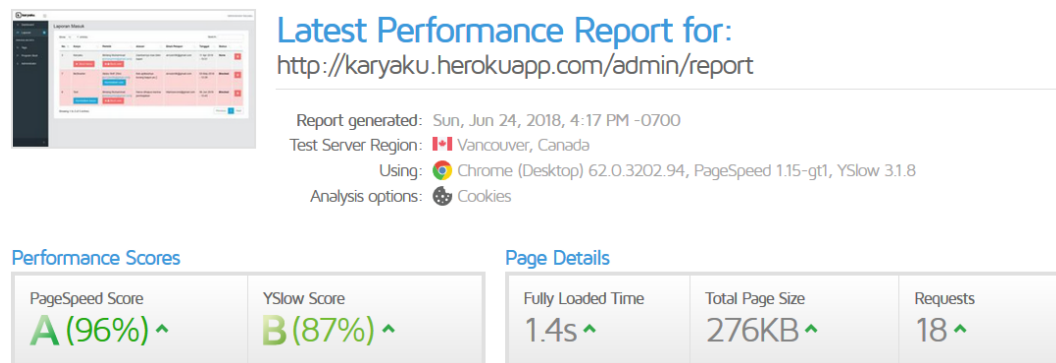
Gambar 111. Hasil Pengujian *Performance Efficiency* Halaman Login Admin

### 13. Halaman *Dashboard*



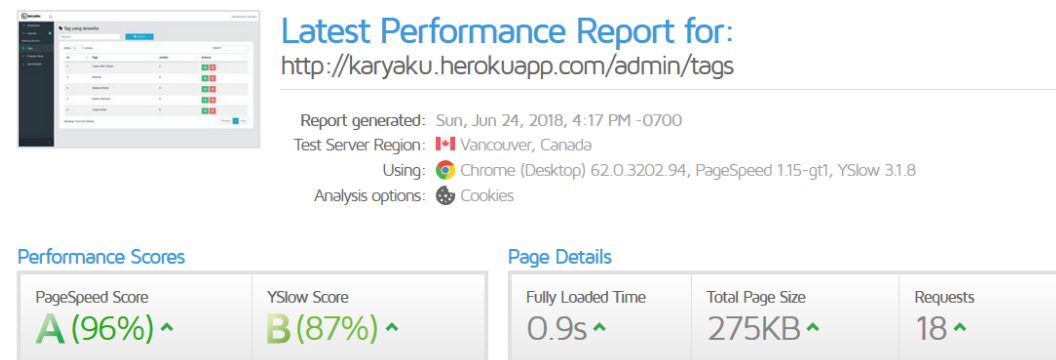
Gambar 112. Hasil Pengujian *Performance Efficiency* Halaman Dashboard Admin

### 14. Halaman Kelola Laporan



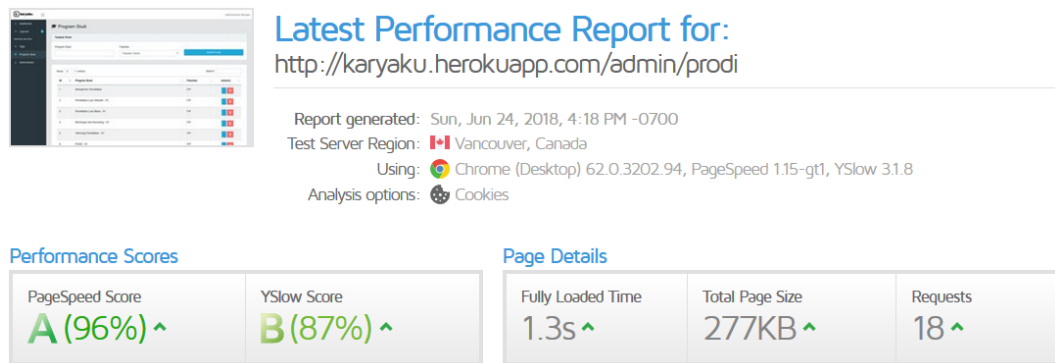
Gambar 113. Hasil Pengujian *Performance Efficiency* Halaman Kelola Laporan

### 15. Halaman Kelola Tag



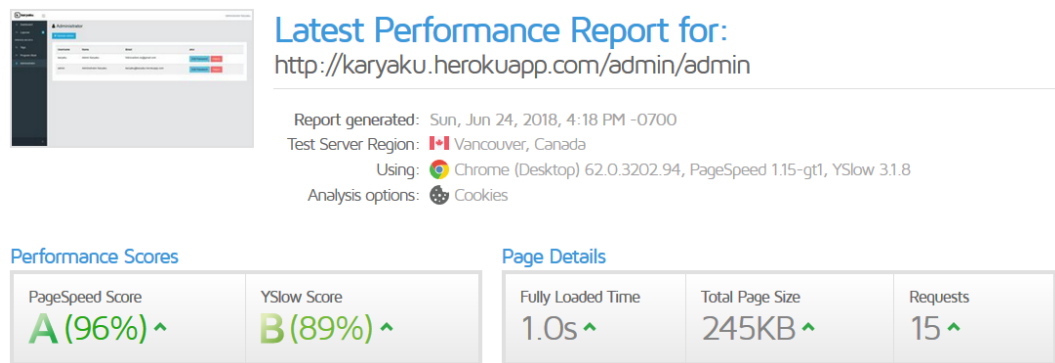
Gambar 114. Hasil Pengujian *Performance Efficiency* Halaman Kelola Tag

## 16. Halaman Kelola Prodi



Gambar 115. Hasil Pengujian *Performance Efficiency* Halaman Kelola Prodi

## 17. Halaman Kelola Administrator



Gambar 116. Hasil Pengujian *Performance Efficiency* Halaman Kelola Administrator

Lampiran 13. Foto Kegiatan

